Python: module spherepack

spherepack index
/pcmdi/halliday1/cdat-4.0/lib/python2.4/site-packages/spherepack_dir/spherepack.so

Fortran interface module spherepack

Functions

divec(...)

```
    ***********************************************************************

    given the vector spherical harmonic coefficients br and bi, precom
    by subroutine vhaec for a vector field (v,w), subroutine divec
    computes the divergence of the vector field in the scalar array dv
    dv(i,j) is the divergence at the colatitude

           theta(i) = (i-1)*pi/(nlat-1)

    and east longitude

           lambda(j) = (j-1)*2*pi/nlon

    on the sphere.  i.e.

           dv(i,j) = 1/sint*[ d(sint*v(i,j))/dtheta + d(w(i,j))/dlambd

    where sint = sin(theta(i)).  w is the east longitudinal and v
    is the colatitudinal component of the vector field from which
    br,bi were precomputed.  required associated legendre polynomials
    are recomputed rather than stored as they are in subroutine dives.

    ************************************************************
    input parameters

    nlat   the number of  colatitudes on the full sphere including the

    nlon   the number of distinct longitude points.


    isym   a parameter which determines whether the divergence is
           computed on the full or half sphere as follows:

     = 0
           the divergence is computed on the entire
           sphere.  i.e., in the array dv(i,j) for i=1,...,nlat and
           j=1,...,nlon.
```

```
nt     nt is the number of scalar and vector fields.

idv    the first dimension of the array dv as it appears in
       the program that calls divec.

jdv    the second dimension of the array dv as it appears in
       the program that calls divec.

br,bi  two or three dimensional arrays (see input parameter nt)
       that contain vector spherical harmonic coefficients
       of the vector field (v,w) as computed by subroutine vhaec.
       br and bi must be computed by vhaec prior to calling
       divec.

mdb    the first dimension of the arrays br and bi as it
       appears in the program that calls divec.

ndb    the second dimension of the arrays br and bi as it
       appears in the program that calls divec.

wshsec an array which must be initialized by subroutine shseci.

lshsec the dimension of the array wshsec as it appears in the
       program that calls divec.

work   a work array that does not have to be saved.

lwork  the dimension of the array work as it appears in the
       program that calls divec.

**************************************************************

output parameters

dv     a two or three dimensional array (see input parameter nt)
      that contains the divergence of the vector field (v,w)
      whose coefficients br,bi where computed by subroutine
      vhaec.

ierror = 0  no errors
       = 1  error in the specification of nlat
       = 2  error in the specification of nlon
       = 3  error in the specification of isym
       = 4  error in the specification of nt
       = 5  error in the specification of idv
       = 6  error in the specification of jdv
       = 7  error in the specification of mdb
       = 8  error in the specification of ndb
       = 9  error in the specification of lshsec
       = 10 error in the specification of lwork
**************************************************************
```

dives(...)

```
     ************************************************************

     given the vector spherical harmonic coefficients br and bi, precom
     by subroutine vhaes for a vector field (v,w), subroutine dives
     computes the divergence of the vector field in the scalar array dv
     dv(i,j) is the divergence at the colatitude

            theta(i) = (i-1)*pi/(nlat-1)

     and east longitude

            lambda(j) = (j-1)*2*pi/nlon

     on the sphere.  i.e.

            dv(i,j) = 1/sint*[ d(sint*v(i,j))/dtheta + d(w(i,j))/dlambd

     where sint = sin(theta(i)).  w is the east longitudinal and v
     is the colatitudinal component of the vector field from which
     br,bi were precomputed.  required associated legendre polynomials
     are recomputed rather than stored as they are in subroutine dives.

     ************************************************************
     input parameters

     nlat   the number of  colatitudes on the full sphere including the

     nlon   the number of distinct longitude points.


     isym   a parameter which determines whether the divergence is
            computed on the full or half sphere as follows:

      = 0
            the divergence is computed on the entire
            sphere.  i.e., in the array dv(i,j) for i=1,...,nlat and
            j=1,...,nlon.

     nt     nt is the number of scalar and vector fields.

     idv    the first dimension of the array dv as it appears in
            the program that calls dives.

     jdv    the second dimension of the array dv as it appears in
            the program that calls dives.

     br,bi  two or three dimensional arrays (see input parameter nt)
            that contain vector spherical harmonic coefficients
            of the vector field (v,w) as computed by subroutine vhaes.
            br and bi must be computed by vhaes prior to calling
            dives.
```

```
      mdb    the first dimension of the arrays br and bi as it
             appears in the program that calls dives.

      ndb    the second dimension of the arrays br and bi as it
             appears in the program that calls dives.

      wshses an array which must be initialized by subroutine shsesi.

      lshses the dimension of the array wshses as it appears in the
             program that calls dives.

      work   a work array that does not have to be saved.

      lwork  the dimension of the array work as it appears in the
             program that calls dives.

      ****************************************************************

      output parameters

      dv     a two or three dimensional array (see input parameter nt)
             that contains the divergence of the vector field (v,w)
             whose coefficients br,bi where computed by subroutine
             vhaes.

      ierror = 0  no errors
             = 1  error in the specification of nlat
             = 2  error in the specification of nlon
             = 3  error in the specification of isym
             = 4  error in the specification of nt
             = 5  error in the specification of idv
             = 6  error in the specification of jdv
             = 7  error in the specification of mdb
             = 8  error in the specification of ndb
             = 9  error in the specification of lshses
             = 10 error in the specification of lwork
      ****************************************************************
```

divgc(...)

```
      ****************************************************************

      given the vector spherical harmonic coefficients br and bi, precom
      by subroutine vhagc for a vector field (v,w), subroutine divgc
      computes the divergence of the vector field in the scalar array dv
      dv(i,j) is the divergence at the gaussian colatitude point theta(i
      (see nlat as input parameter) and east longitude

             lambda(j) = (j-1)*2*pi/nlon

      on the sphere.  i.e.
```

```
       dv(i,j) = 1/sint*[ d(sint*v(i,j))/dtheta + d(w(i,j))/dlambd

where sint = sin(theta(i)).  w is the east longitudinal and v
is the colatitudinal component of the vector field from which
br,bi were precomputed.  required associated legendre polynomials
are recomputed rather than stored as they are in subroutine divgs.

*****************************************************************

input parameters

nlat   the number of points in the gaussian colatitude grid on the
       full sphere.

nlon   the number of distinct londitude points.

isym   a parameter which determines whether the divergence is
       computed on the full or half sphere as follows:

 = 0

       the divergence is computed on the entire sphere.
       i.e., in the array dv(i,j) for i=1,...,nlat and j=1,...,nlo

nt     nt is the number of scalar and vector fields.

idv    the first dimension of the array dv as it appears in
       the program that calls divgc.

jdv    the second dimension of the array dv as it appears in
       the program that calls divgc.

br,bi  two or three dimensional arrays (see input parameter nt)
       that contain vector spherical harmonic coefficients
       of the vector field (v,w) as computed by subroutine vhagc.
       br and bi must be computed by vhagc prior to calling
       divgc.

mdb    the first dimension of the arrays br and bi as it
       appears in the program that calls divgc.

ndb    the second dimension of the arrays br and bi as it
       appears in the program that calls divgc.


wshsgc an array which must be initialized by subroutine shsgci.

lshsgc the dimension of the array wshsgc as it appears in the
       program that calls divgc.

work   a work array that does not have to be saved.
```

```
      lwork  the dimension of the array work as it appears in the
             program that calls divgc.

      ****************************************************************

      output parameters

      dv     a two or three dimensional array (see input parameter nt)
             that contains the divergence of the vector field (v,w)
             whose coefficients br,bi where computed by subroutine
             vhagc.

      ierror= 0  no errors
            = 1  error in the specification of nlat
            = 2  error in the specification of nlon
            = 3  error in the specification of isym
            = 4  error in the specification of nt
            = 5  error in the specification of idv
            = 6  error in the specification of jdv
            = 7  error in the specification of mdb
            = 8  error in the specification of ndb
            = 9  error in the specification of lshsgc
            = 10 error in the specification of lwork
      ****************************************************************
```

divgs(...)

```
      ****************************************************************

      given the vector spherical harmonic coefficients br and bi, precom
      by subroutine vhags for a vector field (v,w), subroutine divgs
      computes the divergence of the vector field in the scalar array di
      divg(i,j) is the divergence at the gaussian colatitude point theta
      (see nlat as input parameter) and east longitude

         lambda(j) = (j-1)*2*pi/nlon

      on the sphere.  i.e.

         dv(i,j) = 1/sint*[ d(sint*v(i,j))/dtheta + d(w(i,j))/dlambda ]

      where sint = sin(theta(i)).  w is the east longitudinal and v
      is the colatitudinal component of the vector field from which
      br,bi were precomputed

      ****************************************************************

      input parameters

      nlat   the number of points in the gaussian colatitude grid on the
      full sphere.

      nlon   the number of distinct longitude points.
```

```
isym   a parameter which determines whether the divergence is
       computed on the full or half sphere as follows:

 = 0

       the divergence is computed on the entire.
       i.e., in the array divg(i,j) for i=1,...,nlat and j=1,...,n

nt     nt is the number of scalar and vector fields.

idiv   the first dimension of the array divg as it appears in
       the program that calls divgs.

jdiv   the second dimension of the array divg as it appears in
       the program that calls divgs.

br,bi  two or three dimensional arrays (see input parameter nt)
       that contain vector spherical harmonic coefficients
       of the vector field (v,w) as computed by subroutine vhags.
       br and bi must be computed by vhags prior to calling
       divgs.

mdb    the first dimension of the arrays br and bi as it
       appears in the program that calls divgs.

ndb    the second dimension of the arrays br and bi as it
        appears in the program that calls divgs.

wshsgs an array which must be intialized by subroutine shsgsi.

lshsgs the dimension of the array wshsgs as it appears in the
       program that calls divgs.

work   a work array that does not have to be saved.

lwork  the dimension of the array work as it appears in the
       program that calls divgs.

*************************************************************

output parameters

divg   a two or three dimensional array (see input parameter nt)
  that contains the divergence of the vector field (v,w)
  whose coefficients br,bi where computed by subroutine
  vhags.

ierror = 0  no errors
       = 1  error in the specification of nlat
       = 2  error in the specification of nlon
       = 3  error in the specification of isym
```

```
           = 4  error in the specification of nt
           = 5  error in the specification of idiv
           = 6  error in the specification of jdiv
           = 7  error in the specification of mdb
           = 8  error in the specification of ndb
           = 9  error in the specification of lshsgs
           = 10 error in the specification of lwork
     ****************************************************************
```

gaqd(...)

```
     ****************************************************************

       subroutine gaqd computes the nlat gaussian colatitudes and weigh
       in doubleprecision. the colatitudes are in radians and lie in th
       in the interval (0,pi).

     ****************************************************************

       input parameters

       nlat    the number of gaussian colatitudes in the interval (0,pi

       dwork   a temporary work space

       ldwork  the length of the work space  in the routine calling gaq
               ldwork must be at least nlat*(nlat+2).

     ****************************************************************

       output parameters

       theta   a doubleprecision vector of length nlat containing the
               nlat gaussian colatitudes on the sphere in increasing ra
               in the interval (o,pi).

       wts     a doubleprecision vector of length nlat containing the
               nlat gaussian weights.

       ierror = 0 no errors
              = 1 if ldwork.lt.nlat*(nlat+2)
              = 2 if nlat.le.0
              = 3 if unable to compute gaussian points
                  (failure in in eigenvalue routine)
     ****************************************************************
```

gradec(...)

```
     ****************************************************************

     given the scalar spherical harmonic coefficients a and b, precompu
     by subroutine shaec for a scalar field sf, subroutine gradec compu
     an irrotational vector field (v,w) such that
```

```
      gradient(sf) = (v,w).

v is the colatitudinal and w is the east longitudinal component
of the gradient.  i.e.,

       v(i,j) = d(sf(i,j))/dtheta

and

       w(i,j) = 1/sint*d(sf(i,j))/dlambda

at colatitude

       theta(i) = (i-1)*pi/(nlat-1)

and longitude

       lambda(j) = (j-1)*2*pi/nlon.

where sint = sin(theta(i)).  required associated legendre polynomi
are recomputed rather than stored as they are in subroutine grades
saves storage (compare wvhsec here and wvhses in grades) but incre
computational requirements.

************************************************************

input parameters

nlat   the number of colatitudes on the full sphere including the
       poles.

nlon   the number of distinct longitude points.

isym   this has the same value as the isym that was input to
       subroutine shaec to compute the arrays a and b from the
       scalar field sf.  isym determines whether (v,w) are
       computed on the full or half sphere as follows:

 = 0

      sf is not symmetric about the equator. in this case
      the vector field (v,w) is computed on the entire sphere.

nt     nt is the number of scalar and vector fields.

idvw   the first dimension of the arrays v,w as it appears in
       the program that calls gradec.

jdvw   the second dimension of the arrays v,w as it appears in
       the program that calls gradec.

a,b    two or three dimensional arrays (see input parameter nt)
```

```
            that contain scalar spherical harmonic coefficients
            of the scalar field array sf as computed by subroutine shae
            a,b must be computed by shaec prior to calling gradec.

      mdab  the first dimension of the arrays a and b as it appears in
            the program that calls gradec (and shaec).

      ndab  the second dimension of the arrays a and b as it appears in
            the program that calls gradec (and shaec).

      wvhsec an array which must be initialized by subroutine vhseci.

      lvhsec the dimension of the array wvhsec as it appears in the
            program that calls gradec.

      work  a work array that does not have to be saved.

      lwork the dimension of the array work as it appears in the
            program that calls gradec.

      ************************************************************

      output parameters

      v,w   two or three dimensional arrays (see input parameter nt) tha
            contain an irrotational vector field such that the gradient
            the scalar field sf is (v,w).

      ierror= 0  no errors
            = 1  error in the specification of nlat
            = 2  error in the specification of nlon
            = 3  error in the specification of isym
            = 4  error in the specification of nt
            = 5  error in the specification of idvw
            = 6  error in the specification of jdvw
            = 7  error in the specification of mdab
            = 8  error in the specification of ndab
            = 9  error in the specification of lvhsec
            = 10 error in the specification of lwork
      ************************************************************
```

grades(...)

```
      ************************************************************

      given the scalar spherical harmonic coefficients a and b, precompu
      by subroutine shaes for a scalar field sf, subroutine grades compu
      an irrotational vector field (v,w) such that

            gradient(sf) = (v,w).

      v is the colatitudinal and w is the east longitudinal component
      of the gradient.  i.e.,
```

```
       v(i,j) = d(sf(i,j))/dtheta

and

       w(i,j) = 1/sint*d(sf(i,j))/dlambda

at colatitude

       theta(i) = (i-1)*pi/(nlat-1)

and longitude

       lambda(j) = (j-1)*2*pi/nlon.

where sint = sin(theta(i)).  required associated legendre polynomi
are recomputed rather than stored as they are in subroutine grades
saves storage (compare wvhses here and wvhses in grades) but incre
computational requirements.

*************************************************************

input parameters

nlat   the number of colatitudes on the full sphere including the
       poles.

nlon   the number of distinct longitude points.

isym   this has the same value as the isym that was input to
       subroutine shaes to compute the arrays a and b from the
       scalar field sf.  isym determines whether (v,w) are
       computed on the full or half sphere as follows:

 = 0

      sf is not symmetric about the equator. in this case
      the vector field (v,w) is computed on the entire sphere.

nt     nt is the number of scalar and vector fields.

idvw   the first dimension of the arrays v,w as it appears in
       the program that calls grades.

jdvw   the second dimension of the arrays v,w as it appears in
       the program that calls grades.

a,b    two or three dimensional arrays (see input parameter nt)
       that contain scalar spherical harmonic coefficients
       of the scalar field array sf as computed by subroutine shae
       a,b must be computed by shaes prior to calling grades.
```

```
      mdab   the first dimension of the arrays a and b as it appears in
             the program that calls grades (and shaes).

      ndab   the second dimension of the arrays a and b as it appears in
             the program that calls grades (and shaes).

      wvhses an array which must be initialized by subroutine vhsesi.

      lvhses the dimension of the array wvhses as it appears in the
             program that calls grades.

      work   a work array that does not have to be saved.

      lwork  the dimension of the array work as it appears in the
             program that calls grades.

      ************************************************************

      output parameters

      v,w    two or three dimensional arrays (see input parameter nt) tha
             contain an irrotational vector field such that the gradient
             the scalar field sf is (v,w).

      ierror= 0  no errors
            = 1  error in the specification of nlat
            = 2  error in the specification of nlon
            = 3  error in the specification of isym
            = 4  error in the specification of nt
            = 5  error in the specification of idvw
            = 6  error in the specification of jdvw
            = 7  error in the specification of mdab
            = 8  error in the specification of ndab
            = 9  error in the specification of lvhses
            = 10 error in the specification of lwork
      ************************************************************
```

gradgc(...)

```
      ************************************************************

      given the scalar spherical harmonic coefficients a and b, precompu
      by subroutine shagc for a scalar field sf, subroutine gradgc compu
      an irrotational vector field (v,w) such that

            gradient(sf) = (v,w).

      v is the colatitudinal and w is the east longitudinal component
      of the gradient.  i.e.,

             v(i,j) = d(sf(i,j))/dtheta

      and
```

```
       w(i,j) = 1/sint*d(sf(i,j))/dlambda

at the gaussian colatitude point theta(i) (see nlat as input
parameter) and longitude lambda(j) = (j-1)*2*pi/nlon where
where sint = sin(theta(i)).  required associated legendre polynomi
are recomputed rather than stored as they are in subroutine gradgs
saves storage (compare lsav with lsav in gradgs) but increases
computational requirements.

*************************************************************

input parameters

nlat   the number of points in the gaussian colatitude grid on the
       full sphere.

nlon   the number of distinct londitude points.

isym   this has the same value as the isym that was input to
       subroutine shagc to compute the arrays a and b from the
       scalar field sf.

 = 0

      sf is not symmetric about the equator. in this case
      the vector field (v,w) is computed on the entire sphere.

nt     nt is the number of scalar and vector fields.

idvw   the first dimension of the arrays v,w as it appears in
       the program that calls gradgc.

jdvw   the second dimension of the arrays v,w as it appears in
       the program that calls gradgc.

a,b    two or three dimensional arrays (see input parameter nt)
       that contain scalar spherical harmonic coefficients
       of the scalar field array sf as computed by subroutine shag
       a,b must be computed by shagc prior to calling gradgc.

mdab   the first dimension of the arrays a and b as it appears in
       the program that calls gradgc (and shagc).

ndab   the second dimension of the arrays a and b as it appears in
       the program that calls gradgc (and shagc).

wvhsgc an array which must be initialized by subroutine vhsgci.

lvhsgc the dimension of the array wvhsgc as it appears in the
       program that calls gradgc. Let
```

```
      work   a work array that does not have to be saved.

      lwork  the dimension of the array work as it appears in the
             program that calls gradgc. define

      ************************************************************

      output parameters

      v,w    two or three dimensional arrays (see input parameter nt) tha
             contain an irrotational vector field such that the gradient
             the scalar field sf is (v,w).

      ierror= 0  no errors
            = 1  error in the specification of nlat
            = 2  error in the specification of nlon
            = 3  error in the specification of isym
            = 4  error in the specification of nt
            = 5  error in the specification of idvw
            = 6  error in the specification of jdvw
            = 7  error in the specification of mdab
            = 8  error in the specification of ndab
            = 9  error in the specification of lvhsgc
            = 10 error in the specification of lwork
      ************************************************************
```

gradgs(...)

```
      ************************************************************

      given the scalar spherical harmonic coefficients a and b, precompu
      by subroutine shags for a scalar field sf, subroutine gradgs compu
      an irrotational vector field (v,w) such that

            gradient(sf) = (v,w).

      v is the colatitudinal and w is the east longitudinal component
      of the gradient.  i.e.,

             v(i,j) = d(sf(i,j))/dtheta

      and

             w(i,j) = 1/sint*d(sf(i,j))/dlambda

      at the gaussian colatitude point theta(i) (see nlat as input
      parameter) and longitude lambda(j) = (j-1)*2*pi/nlon where
      where sint = sin(theta(i)).  required associated legendre polynomi
      are recomputed rather than stored as they are in subroutine gradgs
      saves storage (compare lsav with lsav in gradgs) but increases
      computational requirements.

      ************************************************************
```

```
input parameters

nlat   the number of points in the gaussian colatitude grid on the
       full sphere.

nlon   the number of distinct londitude points.

isym   this has the same value as the isym that was input to
       subroutine shags to compute the arrays a and b from the
       scalar field sf.

 = 0

      sf is not symmetric about the equator. in this case
      the vector field (v,w) is computed on the entire sphere.

nt     nt is the number of scalar and vector fields.

idvw   the first dimension of the arrays v,w as it appears in
       the program that calls gradgs.

jdvw   the second dimension of the arrays v,w as it appears in
       the program that calls gradgs.

a,b    two or three dimensional arrays (see input parameter nt)
       that contain scalar spherical harmonic coefficients
       of the scalar field array sf as computed by subroutine shag
       a,b must be computed by shags prior to calling gradgs.

mdab   the first dimension of the arrays a and b as it appears in
       the program that calls gradgs (and shags).

ndab   the second dimension of the arrays a and b as it appears in
       the program that calls gradgs (and shags).

wvhsgs an array which must be initialized by subroutine vhsgsi.

lvhsgs the dimension of the array wvhsgs as it appears in the
       program that calls gradgs. Let

work   a work array that does not have to be saved.

lwork  the dimension of the array work as it appears in the
       program that calls gradgs. define

************************************************************

output parameters

v,w    two or three dimensional arrays (see input parameter nt) tha
      contain an irrotational vector field such that the gradient
```

```
            the scalar field sf is (v,w).

      ierror= 0  no errors
            = 1  error in the specification of nlat
            = 2  error in the specification of nlon
            = 3  error in the specification of isym
            = 4  error in the specification of nt
            = 5  error in the specification of idvw
            = 6  error in the specification of jdvw
            = 7  error in the specification of mdab
            = 8  error in the specification of ndab
            = 9  error in the specification of lvhsgs
            = 10 error in the specification of lwork
      ****************************************************************
```

hrfftb(...)

```
      ****************************************************************

        subroutine hrfftb computes the real perodic sequence of m
        sequences from their fourier coefficients (fourier synthesis).
        the transform is defined below at output parameter r.

      ****************************************************************

        input parameters

        m       the number of sequences.

        n       the length of all m sequences.  the method is most
                efficient when n is a product of small primes. n may
                change as long as different work arrays are provided

        r       r(m,n) is a two dimensional real array that contains
                the fourier coefficients of m sequences each with
                length n.

        mdimr   the first dimension of the r array as it appears
                in the program that calls hrfftb. mdimr must be
                greater than or equal to m.

        wsave   a work array which must be dimensioned at least 2*n+15.
                in the program that calls hrfftb. the wsave array must b
                initialized by calling subroutine hrffti(n,wsave) and a
                different wsave array must be used for each different
                value of n. this initialization does not have to be
                repeated so long as n remains unchanged thus subsequent
                transforms can be obtained faster than the first.

        work    a real work array with m*n locations.
```

```
*****************************************************************

  output parameters

  r      for all j=1,...,m

           r(j,1) = the sum from i=1 to i=n of r(j,i)

  wsave   contains results which must not be destroyed between
          calls of hrfftb or hrfftf.

  work    a real work array with m*n locations that does not
          have to be saved

*****************************************************************
```

hrfftf(...)

```
*****************************************************************

  subroutine hrfftf computes the fourier coefficients of m real
  perodic sequences (fourier analysis); i.e. hrfftf computes the
  real fft of m sequences each with length n. the transform is
  defined below at output parameter r.

*****************************************************************

  input parameters

  m       the number of sequences.

  n       the length of all m sequences.  the method is most
          efficient when n is a product of small primes. n may
          change as long as different work arrays are provided

  r       r(m,n) is a two dimensional real array that contains m
          sequences each with length n.

  mdimr   the first dimension of the r array as it appears
          in the program that calls hrfftf. mdimr must be
          greater than or equal to m.


  wsave   a work array with at least least 2*n+15 locations
          in the program that calls hrfftf. the wsave array must b
          initialized by calling subroutine hrffti(n,wsave) and a
          different wsave array must be used for each different
          value of n. this initialization does not have to be
          repeated so long as n remains unchanged thus subsequent
          transforms can be obtained faster than the first.
          the same wsave array can be used by hrfftf and hrfftb.

  work    a real work array with m*n locations.
```

```
*****************************************************************

  output parameters

  r      for all j=1,...,m

          r(j,1) = the sum from i=1 to i=n of r(j,i)


  wsave   contains results which must not be destroyed between
          calls of hrfftf or hrfftb.

  work    a real work array with m*n locations that does
          not have to be saved.
*****************************************************************
```

hrffti(...)

```
*****************************************************************

  subroutine hrffti initializes the array wsave which is used in
  both hrfftf and hrfftb. the prime factorization of n together
  with a tabulation of the trigonometric functions are computed an
  stored in wsave.

*****************************************************************

  input parameter

  n       the length of the sequence to be transformed.


*****************************************************************

  output parameter

  wsave   a work array which must be dimensioned at least 2*n+15.
          the same work array can be used for both hrfftf and
          hrfftb as long as n remains unchanged. different wsave
          arrays are required for different values of n. the
          contents of wsave must not be changed between calls
          of hrfftf or hrfftb.

*****************************************************************
```

idivec(...)

```
*****************************************************************

given the scalar spherical harmonic coefficients a and b, precompu
by subroutine shaec for a scalar array dv, subroutine idivec compu
an irrotational vector field (v,w) whose divergence is dv - pertrb
w is the east longitude component and v is the colatitudinal compo
```

```
pertrb is a constant which must be subtracted from dv for (v,w) to
exist (see the description of pertrb below).  usually pertrb is ze
or small relative to dv.  the vorticity of (v,w), as computed by
vortec, is the zero scalar field.  v(i,j) and w(i,j) are the
velocity components at colatitude

       theta(i) = (i-1)*pi/(nlat-1)

and longitude

       lambda(j) = (j-1)*2*pi/nlon.

the
       divergence[v(i,j),w(i,j)]

     = [d(w(i,j)/dlambda + d(sint*v(i,j))/dtheta]/sint

     = dv(i,j) - pertrb

and

       vorticity(v(i,j),w(i,j))

    =  [dv/dlambda - d(sint*w)/dtheta]/sint

    =  0.0

where sint = sin(theta(i)).  required associated legendre polynomi
are recomputed rather than stored as they are in subroutine idives

*************************************************************

input parameters

nlat   the number of colatitudes on the full sphere including the
       poles.

nlon   the number of distinct londitude points.


isym   this has the same value as the isym that was input to
       subroutine shaec to compute the arrays a and b from the
       scalar field dv.  isym determines whether (v,w) are
       computed on the full or half sphere as follows:

 = 0

      dv is not symmetric about the equator. in this case
      the vector field (v,w) is computed on the entire sphere.

nt     nt is the number of divergence and vector fields.
```

```
idvw   the first dimension of the arrays v,w as it appears in
       the program that calls idivec. if isym = 0 then idvw
       must be at least nlat.

jdvw   the second dimension of the arrays v,w as it appears in
       the program that calls idivec.

a,b    two or three dimensional arrays (see input parameter nt)
       that contain scalar spherical harmonic coefficients
       of the divergence array dv as computed by subroutine shaec.
       a,b must be computed by shaec prior to calling idivec.

mdab   the first dimension of the arrays a and b as it appears in
       the program that calls idivec (and shaec).

ndab   the second dimension of the arrays a and b as it appears in
       the program that calls idivec (and shaec).

wvhsec an array which must be initialized by subroutine vhseci.

lvhsec the dimension of the array wvhsec as it appears in the
       program that calls idivec.

work   a work array that does not have to be saved.

lwork  the dimension of the array work as it appears in the
       program that calls idivec.

************************************************************

output parameters
v,w    two or three dimensional arrays (see input parameter nt) tha
      contain an irrotational vector field whose divergence is
      dv-pertrb.

pertrb  a nt dimensional array (see input parameter nt and assume
      for the description that follows).  dv - pertrb is a scalar
      field which can be the divergence of a vector field (v,w).
      pertrb is related to the scalar harmonic coefficients a,b
      of dv (computed by shaec) by the formula

           pertrb = a(1,1)/(2.*sqrt(2.))

ierror = 0  no errors
       = 1  error in the specification of nlat
       = 2  error in the specification of nlon
       = 3  error in the specification of isym
       = 4  error in the specification of nt
       = 5  error in the specification of idvw
       = 6  error in the specification of jdvw
       = 7  error in the specification of mdab
       = 8  error in the specification of ndab
```

```
           = 9  error in the specification of lvhsec
           = 10 error in the specification of lwork
     ****************************************************************
```

idives(...)

```
     ****************************************************************

     given the scalar spherical harmonic coefficients a and b, precompu
     by subroutine shaes for a scalar array dv, subroutine idives compu
     an irrotational vector field (v,w) whose divergence is dv - pertrb
     w is the east longitude component and v is the colatitudinal compo
     pertrb is a constant which must be subtracted from dv for (v,w) to
     exist (see the description of pertrb below).  usually pertrb is ze
     or small relative to dv.  the vorticity of (v,w), as computed by
     vortes, is the zero scalar field.  i.e., v(i,j) and w(i,j) are the
     velocity components at colatitude

            theta(i) = (i-1)*pi/(nlat-1)

     and longitude

            lambda(j) = (j-1)*2*pi/nlon.

     the

            divergence[v(i,j),w(i,j)]

          = [d(w(i,j)/dlambda + d(sint*v(i,j))/dtheta]/sint

          = dv(i,j) - pertrb

     and

            vorticity(v(i,j),w(i,j))

         =  [dv/dlambda - d(sint*w)/dtheta]/sint

         =  0.0

     where sint = sin(theta(i)).  required associated legendre polynomi
     are stored rather than recomputed as they are in subroutine idivec

     ****************************************************************

     input parameters

     nlat   the number of colatitudes on the full sphere including the
            poles.

     nlon   the number of distinct londitude points.

     isym   this has the same value as the isym that was input to
```

subroutine shaes to compute the arrays a and b from the scalar field dv. isym determines whether (v,w) are computed on the full or half sphere as follows:

= 0
dv is not symmetric about the equator. in this case the vector field (v,w) is computed on the entire sphere.

nt nt is the number of divergence and vector fields.

idvw the first dimension of the arrays v,w as it appears in the program that calls idives.

jdvw the second dimension of the arrays v,w as it appears in the program that calls idives.

a,b two or three dimensional arrays (see input parameter nt) that contain scalar spherical harmonic coefficients of the divergence array dv as computed by subroutine shaes. a,b must be computed by shaes prior to calling idives.

mdab the first dimension of the arrays a and b as it appears in the program that calls idives (and shaes).

ndab the second dimension of the arrays a and b as it appears in the program that calls idives (and shaes).

wvhses an array which must be initialized by subroutine vhesesi.

lvhses the dimension of the array wvhses as it appears in the program that calls idives.

work a work array that does not have to be saved.

lwork the dimension of the array work as it appears in the program that calls idives.

****

output parameters

v,w two or three dimensional arrays (see input parameter nt) tha contain an irrotational vector field whose divergence is dv-pertrb.

pertrb a nt dimensional array (see input parameter nt and assume for the description that follows). dv - pertrb is a scalar field which can be the divergence of a vector field (v,w). pertrb is related to the scalar harmonic coefficients a,b of dv (computed by shaes) by the formula

```
                pertrb = a(1,1)/(2.*sqrt(2.))

      ierror = 0  no errors
             = 1  error in the specification of nlat
             = 2  error in the specification of nlon
             = 3  error in the specification of isym
             = 4  error in the specification of nt
             = 5  error in the specification of idvw
             = 6  error in the specification of jdvw
             = 7  error in the specification of mdab
             = 8  error in the specification of ndab
             = 9  error in the specification of lvhses
             = 10 error in the specification of lwork
      ****************************************************************
```

idivgc(...)

```
      ****************************************************************

      given the scalar spherical harmonic coefficients a and b, precompu
      by subroutine shagc for a scalar array dv, subroutine idivgc compu
      an irrotational vector field (v,w) whose divergence is dv - pertrb
      w is the east longitude component and v is the colatitudinal compo
      pertrb is a constant which must be subtracted from dv for (v,w) to
      exist (see the description of pertrb below).  usually pertrb is ze
      or small relative to dv.  the vorticity of (v,w) is the zero scala
      field.  v(i,j) and w(i,j) are the velocity components at the gauss
      colatitude theta(i) (see nlat) and longitude lambda(j)=(j-1)*2*pi/
      the

             divergence[v(i,j),w(i,j)]

           = [d(w(i,j)/dlambda + d(sint*v(i,j))/dtheta]/sint

           = dv(i,j) - pertrb

      and

             vorticity(v(i,j),w(i,j))

          =  [dv/dlambda - d(sint*w)/dtheta]/sint

          =  0.0
      where sint = sin(theta(i)).

      ****************************************************************

      input parameters

      nlat   the number of points in the gaussian colatitude grid on the
             full sphere.

      nlon   the number of distinct londitude points.
```

```
isym   this has the same value as the isym that was input to
       subroutine shagc to compute the arrays a and b from the
       scalar field dv.  isym determines whether (v,w) are
       computed on the full or half sphere as follows:

 = 0

      dv is not symmetric about the equator. in this case
      the vector field (v,w) is computed on the entire sphere.

nt     nt is the number of divergence and vector fields.

idvw   the first dimension of the arrays v,w as it appears in
       the program that calls idivgc.

jdvw   the second dimension of the arrays v,w as it appears in
       the program that calls idivgc.

a,b    two or three dimensional arrays (see input parameter nt)
       that contain scalar spherical harmonic coefficients
       of the divergence array dv as computed by subroutine shagc.
       a,b must be computed by shagc prior to calling idivgc.

mdab   the first dimension of the arrays a and b as it appears in
       the program that calls idivgc (and shagc).

ndab   the second dimension of the arrays a and b as it appears in
       the program that calls idivgc (and shagc).

wvhsgc an array which must be initialized by subroutine vhsgci.

lvhsgc the dimension of the array wvhsgc as it appears in the
       program that calls idivgc.

work   a work array that does not have to be saved.

lwork  the dimension of the array work as it appears in the
       program that calls idivgc.


**************************************************************

output parameters


v,w    two or three dimensional arrays (see input parameter nt) tha
       contain an irrotational vector field whose divergence is
       dv-pertrb.

pertrb a nt dimensional array (see input parameter nt and assume n
       for the description that follows).  dv - pertrb is a scalar
```

```
      field which can be the divergence of a vector field (v,w).
      pertrb is related to the scalar harmonic coefficients a,b
      of dv (computed by shagc) by the formula

           pertrb = a(1,1)/(2.*sqrt(2.))

ierror = 0  no errors
       = 1  error in the specification of nlat
       = 2  error in the specification of nlon
       = 3  error in the specification of isym
       = 4  error in the specification of nt
       = 5  error in the specification of idvw
       = 6  error in the specification of jdvw
       = 7  error in the specification of mdab
       = 8  error in the specification of ndab
       = 9  error in the specification of lvhsgc
       = 10 error in the specification of lwork
************************************************************
```

idivgs(...)

```
************************************************************


given the scalar spherical harmonic coefficients a and b, precompu
by subroutine shags for a scalar array divg, subroutine idivgs com
an irrotational vector field (v,w) whose divergence is divg - pert
w is the east longitude component and v is the colatitudinal compo
pertrb is a constant which must be subtracted from divg for (v,w)
exist (see the description of pertrb below).  usually pertrb is ze
or small relative to divg.  the vorticity of (v,w) is the zero sca
field.  v(i,j) and w(i,j) are the velocity components at the gauss
colatitude theta(i) (see nlat) and longitude lambda(j)=(j-1)*2*pi/
the

       divergence[v(i,j),w(i,j)]

     = [d(w(i,j)/dlambda + d(sint*v(i,j))/dtheta]/sint

     = divg(i,j) - pertrb

and

       vorticity(v(i,j),w(i,j))

    =  [dv/dlambda - d(sint*w)/dtheta]/sint

    =  0.0

where sint = sin(theta(i)).

************************************************************
```

input parameters

nlat the number of points in the gaussian colatitude grid on the full sphere.

nlon the number of distinct londitude points. nlon determines the grid increment in longitude as 2*pi/nlon. for example nlon = 72 for a five degree grid. nlon must be greater than 3. the efficiency of the computation is improved when nlon is a product of small prime numbers.

isym this has the same value as the isym that was input to subroutine shags to compute the arrays a and b from the scalar field divg. isym determines whether (v,w) are computed on the full or half sphere as follows:

= 0

divg is not symmetric about the equator. in this case the vector field (v,w) is computed on the entire sphere.

nt nt is the number of divergence and vector fields.

idvw the first dimension of the arrays v,w as it appears in the program that calls idivgs.

jdvw the second dimension of the arrays v,w as it appears in the program that calls idivgs.

a,b two or three dimensional arrays (see input parameter nt) that contain scalar spherical harmonic coefficients of the divergence array divg as computed by subroutine shag a,b must be computed by shags prior to calling idivgs.

mdab the first dimension of the arrays a and b as it appears in the program that calls idivgs (and shags).

ndab the second dimension of the arrays a and b as it appears in the program that calls idivgs (and shags).

wvhsgs an array which must be initialized by subroutine vhsgsi. once initialized,
wvhsgs can be used repeatedly by idivgs as long as nlon and nlat remain unchanged. wvhsgs must not be altered between calls of idivgs.

lvhsgs the dimension of the array wvhsgs as it appears in the program that calls idivgs.

```
work   a work array that does not have to be saved.

lwork  the dimension of the array work as it appears in the
       program that calls idivgs. define

*************************************************************

output parameters


v,w    two or three dimensional arrays (see input parameter nt) tha
       contain an irrotational vector field whose divergence is
       divg-pertrb.

pertrb  a nt dimensional array (see input parameter nt and assume
       for the description that follows).  divg - pertrb is a scala
       field which can be the divergence of a vector field (v,w).
       pertrb is related to the scalar harmonic coefficients a,b
       of divg (computed by shags) by the formula

            pertrb = a(1,1)/(2.*sqrt(2.))

       the unperturbed scalar field divg can be the divergence of a
       vector field only if a(1,1) is zero.

ierror = 0  no errors
       = 1  error in the specification of nlat
       = 2  error in the specification of nlon
       = 3  error in the specification of isym
       = 4  error in the specification of nt
       = 5  error in the specification of idvw
       = 6  error in the specification of jdvw
       = 7  error in the specification of mdab
       = 8  error in the specification of ndab
       = 9  error in the specification of lvhsgs
       = 10 error in the specification of lwork
*************************************************************
```

idvtec(...)

```
*************************************************************

given the scalar spherical harmonic coefficients ad,bd precomputed
by subroutine shaec for the scalar field divg and coefficients av,
precomputed by subroutine shaec for the scalar field vort, subrout
idvtec computes a vector field (v,w) whose divergence is divg - pe
and whose vorticity is vort - pertbv.  w the is east longitude com
and v is the colatitudinal component of the velocity.  if nt=1 (se
below) pertrbd and pertbv are constants which must be subtracted f
divg and vort for (v,w) to exist (see the description of pertbd an
pertrbv below).  usually pertbd and pertbv are zero or small relat
to divg and vort.  w(i,j) and v(i,j) are the velocity components a
colatitude
```

```
       theta(i) = (i-1)*pi/(nlat-1)

and longitude

       lambda(j) = (j-1)*2*pi/nlon

the

       divergence(v(i,j),w(i,j))

    =  [d(sint*v)/dtheta + dw/dlambda]/sint


    =  divg(i,j) - pertbd

and

       vorticity(v(i,j),w(i,j))

    =  [-dv/dlambda + d(sint*w)/dtheta]/sint

    =  vort(i,j) - pertbv

where

       sint = cos(theta(i)).


************************************************************

input parameters

nlat   the number of colatitudes on the full sphere including the
       poles.

nlon   the number of distinct longitude points.


isym   isym determines whether (v,w) are computed on the full or h
       sphere as follows:

 = 0
       In this case, the vector field (v,w) is computed on the ent
       sphere.

nt     in the program that calls idvtec, nt is the number of scala
       and vector fields.

idvw   the first dimension of the arrays v,w as it appears in
       the program that calls idvtec.
```

```
jdvw   the second dimension of the arrays v,w as it appears in
       the program that calls idvtec.

ad,bd  two or three dimensional arrays (see input parameter nt)
       that contain scalar spherical harmonic coefficients
       of the divergence array divg as computed by subroutine shae

av,bv  two or three dimensional arrays (see input parameter nt)
       that contain scalar spherical harmonic coefficients
       of the vorticity array vort as computed by subroutine shaec
       ad,bd,av,bv must be computed by shaec prior to calling idvt

mdab   the first dimension of the arrays ad,bd,av,bv as it appears
       in the program that calls idvtec (and shaec).

ndab   the second dimension of the arrays ad,bd,av,bv as it appear
       the program that calls idvtec (and shaec).

wvhse  an array which must be initialized by subroutine vhseci.

lvhse  the dimension of the array wvhsec as it appears in the
       program that calls idvtec.

work   a work array that does not have to be saved.

lwork  the dimension of the array work as it appears in the
       program that calls idvtec.

************************************************************

output parameters


v,w    two or three dimensional arrays (see input parameter nt) tha
       contain a vector field whose divergence is divg - pertbd and
       whose vorticity is vort - pertbv.

pertbd a nt dimensional array (see input parameter nt and assume n
       for the description that follows).  divg - pertbd is a scala
       field which can be the divergence of a vector field (v,w).
       pertbd is related to the scalar harmonic coefficients ad,bd
       of divg (computed by shaec) by the formula

            pertbd = ad(1,1)/(2.*sqrt(2.))

       an unperturbed divg can be the divergence of a vector field
       only if ad(1,1) is zero.

pertbv a nt dimensional array (see input parameter nt and assume n
       for the description that follows).  vort - pertbv is a scala
       field which can be the vorticity of a vector field (v,w).
       pertbv is related to the scalar harmonic coefficients av,bv
```

```
      of vort (computed by shaec) by the formula

           pertbv = av(1,1)/(2.*sqrt(2.))

      an unperturbed vort can be the vorticity of a vector field
      only if av(1,1) is zero.

ierror = 0  no errors
      = 1  error in the specification of nlat
      = 2  error in the specification of nlon
      = 3  error in the specification of isym
      = 4  error in the specification of nt
      = 5  error in the specification of idvw
      = 6  error in the specification of jdvw
      = 7  error in the specification of mdab
      = 8  error in the specification of ndab
      = 9  error in the specification of lvhsec
      = 10 error in the specification of lwork
*****************************************************************
```

idvtes(...)

```
*****************************************************************

given the scalar spherical harmonic coefficients ad,bd precomputed
by subroutine shaes for the scalar field divg and coefficients av,
precomputed by subroutine shaes for the scalar field vort, subrout
idvtes computes a vector field (v,w) whose divergence is divg - pe
and whose vorticity is vort - pertbv.  w the is east longitude com
and v is the colatitudinal component of the velocity.  if nt=1 (se
below) pertrbd and pertbv are constants which must be subtracted f
divg and vort for (v,w) to exist (see the description of pertbd an
pertrbv below).  usually pertbd and pertbv are zero or small relat
to divg and vort.  w(i,j) and v(i,j) are the velocity components a
colatitude

       theta(i) = (i-1)*pi/(nlat-1)

and longitude

       lambda(j) = (j-1)*2*pi/nlon

the

       divergence(v(i,j),w(i,j))

    =  [d(sint*v)/dtheta + dw/dlambda]/sint

    =  divg(i,j) - pertbd

and
```

```
       vorticity(v(i,j),w(i,j))

    =  [-dv/dlambda + d(sint*w)/dtheta]/sint

    =  vort(i,j) - pertbv

where

       sint = cos(theta(i)).

*****************************************************************


input parameters

nlat   the number of colatitudes on the full sphere including the
       poles.

nlon   the number of distinct londitude points.

isym   isym determines whether (v,w) are computed on the full or h
       sphere as follows:

 = 0
       In this case, the vector field (v,w) is computed on the ent
       sphere.

nt     in the program that calls idvtes, nt is the number of scala
       and vector fields.

idvw   the first dimension of the arrays v,w as it appears in
       the program that calls idvtes.

jdvw   the second dimension of the arrays v,w as it appears in
       the program that calls idvtes.

ad,bd  two or three dimensional arrays (see input parameter nt)
       that contain scalar spherical harmonic coefficients
       of the divergence array divg as computed by subroutine shae

av,bv  two or three dimensional arrays (see input parameter nt)
       that contain scalar spherical harmonic coefficients
       of the vorticity array vort as computed by subroutine shaes
       ad,bd,av,bv must be computed by shaes prior to calling idvt

mdab   the first dimension of the arrays ad,bd,av,bv as it appears
       in the program that calls idvtes (and shaes).

ndab   the second dimension of the arrays ad,bd,av,bv as it appear
       the program that calls idvtes (and shaes).

wvhses an array which must be initialized by subroutine vhsesi.
```

```
lvhses the dimension of the array wvhses as it appears in the
       program that calls idvtes. define

work   a work array that does not have to be saved.

lwork  the dimension of the array work as it appears in the
       program that calls idvtes.

************************************************************

output parameters

v,w    two or three dimensional arrays (see input parameter nt) tha
      contain a vector field whose divergence is divg - pertbd and
      whose vorticity is vort - pertbv.

pertbd  a nt dimensional array (see input parameter nt and assume
      for the description that follows).  divg - pertbd is a scala
      field which can be the divergence of a vector field (v,w).
      pertbd is related to the scalar harmonic coefficients ad,bd
      of divg (computed by shaes) by the formula

           pertbd = ad(1,1)/(2.*sqrt(2.))

      an unperturbed divg can be the divergence of a vector field
      only if ad(1,1) is zero.

pertbv a nt dimensional array (see input parameter nt and assume n
      for the description that follows).  vort - pertbv is a scala
      field which can be the vorticity of a vector field (v,w).
      pertbv is related to the scalar harmonic coefficients av,bv
      of vort (computed by shaes) by the formula

           pertbv = av(1,1)/(2.*sqrt(2.))

      an unperturbed vort can be the vorticity of a vector field
      only if av(1,1) is zero.

ierror = 0  no errors
      = 1  error in the specification of nlat
      = 2  error in the specification of nlon
      = 3  error in the specification of isym
      = 4  error in the specification of nt
      = 5  error in the specification of idvw
      = 6  error in the specification of jdvw
      = 7  error in the specification of mdab
      = 8  error in the specification of ndab
      = 9  error in the specification of lvhses
      = 10 error in the specification of lwork
************************************************************
```

```
      ************************************************************

      given the scalar spherical harmonic coefficients ad,bd precomputed
      by subroutine shagc for the scalar field divg and coefficients av,
      precomputed by subroutine shagc for the scalar field vort, subrout
      idvtgc computes a vector field (v,w) whose divergence is divg - pe
      and whose vorticity is vort - pertbv.  w the is east longitude com
      and v is the colatitudinal component of the velocity.  if nt=1 (se
      below) pertrbd and pertbv are constants which must be subtracted f
      divg and vort for (v,w) to exist (see the description of pertbd an
      pertrbv below).  usually pertbd and pertbv are zero or small relat
      to divg and vort.  w(i,j) and v(i,j) are the velocity components a
      gaussian colatitude theta(i) (see nlat as input argument) and long
      lambda(j) = (j-1)*2*pi/nlon

      the

             divergence(v(i,j),w(i,j))

          =  [d(sint*v)/dtheta + dw/dlambda]/sint

          =  divg(i,j) - pertbd

      and

             vorticity(v(i,j),w(i,j))

          =  [-dv/dlambda + d(sint*w)/dtheta]/sint

          =  vort(i,j) - pertbv

      where

             sint = cos(theta(i)).

      ************************************************************

      input parameters

      nlat   the number of points in the gaussian colatitude grid on the
             full sphere.

      nlon   the number of distinct londitude points.

      isym   isym determines whether (v,w) are computed on the full or h
             sphere as follows:

       = 0
             In this case, the vector field (v,w) is computed on the ent
             sphere.
```

```
nt     in the program that calls idvtgc, nt is the number of scala
       and vector fields.

idvw   the first dimension of the arrays v,w as it appears in
       the program that calls idvtgc.

jdvw   the second dimension of the arrays v,w as it appears in
       the program that calls idvtgc.

ad,bd  two or three dimensional arrays (see input parameter nt)
       that contain scalar spherical harmonic coefficients
       of the divergence array divg as computed by subroutine shag

av,bv  two or three dimensional arrays (see input parameter nt)
       that contain scalar spherical harmonic coefficients
       of the vorticity array vort as computed by subroutine shagc
       ad,bd,av,bv must be computed by shagc prior to calling idvt

mdab   the first dimension of the arrays ad,bd,av,bv as it appears
       in the program that calls idvtgc (and shagc).

ndab   the second dimension of the arrays ad,bd,av,bv as it appear
       the program that calls idvtgc (and shagc).

wvhsgc an array which must be initialized by subroutine vhsgci.


lvhsgc the dimension of the array wvhsgc as it appears in the
       program that calls idvtgc.

work   a work array that does not have to be saved.

lwork  the dimension of the array work as it appears in the
       program that calls idvtgc.

**********************************************************

output parameters


v,w    two or three dimensional arrays (see input parameter nt) tha
       contain a vector field whose divergence is divg - pertbd and
       whose vorticity is vort - pertbv.

pertbd  a nt dimensional array (see input parameter nt and assume
       for the description that follows).  divg - pertbd is a scala
       field which can be the divergence of a vector field (v,w).

pertbv a nt dimensional array (see input parameter nt and assume n
       for the description that follows).  vort - pertbv is a scala
       field which can be the vorticity of a vector field (v,w).
```

```
      ierror = 0  no errors
             = 1  error in the specification of nlat
             = 2  error in the specification of nlon
             = 3  error in the specification of isym
             = 4  error in the specification of nt
             = 5  error in the specification of idvw
             = 6  error in the specification of jdvw
             = 7  error in the specification of mdab
             = 8  error in the specification of ndab
             = 9  error in the specification of lvhsgc
             = 10 error in the specification of lwork
      ****************************************************************
```

idvtgs(...)

```
      ****************************************************************

      given the scalar spherical harmonic coefficients ad,bd precomputed
      by subroutine shaes for the scalar field divg and coefficients av,
      precomputed by subroutine shaes for the scalar field vort, subrout
      idvtgs computes a vector field (v,w) whose divergence is divg - pe
      and whose vorticity is vort - pertbv.  w the is east longitude com
      and v is the colatitudinal component of the velocity.  if nt=1 (se
      below) pertrbd and pertbv are constants which must be subtracted f
      divg and vort for (v,w) to exist (see the description of pertbd an
      pertrbv below).  usually pertbd and pertbv are zero or small relat
      to divg and vort.  w(i,j) and v(i,j) are the velocity components a
      gaussian colatitude theta(i) (see nlat as input argument) and long
      lambda(j) = (j-1)*2*pi/nlon

      the

             divergence(v(i,j),w(i,j))

          =  [d(sint*v)/dtheta + dw/dlambda]/sint

          =  divg(i,j) - pertbd

      and

             vorticity(v(i,j),w(i,j))

          =  [-dv/dlambda + d(sint*w)/dtheta]/sint

          =  vort(i,j) - pertbv

      where

             sint = cos(theta(i)).



      ****************************************************************
```

```
input parameters

nlat   the number of points in the gaussian colatitude grid on the
       full sphere.

nlon   the number of distinct longitude points.

isym   isym determines whether (v,w) are computed on the full or h
       sphere as follows:

 = 0
       divg,vort are neither pairwise symmetric/antisymmetric nor
       antisymmetric/symmetric about the equator as described for
       In this case, the vector field (v,w) is computed on the ent
       sphere.

nt     in the program that calls idvtgs, nt is the number of scala
       and vector fields.

idvw   the first dimension of the arrays v,w as it appears in
       the program that calls idvtgs.

jdvw   the second dimension of the arrays v,w as it appears in
       the program that calls idvtgs. jdvw must be at least nlon.

ad,bd  two or three dimensional arrays (see input parameter nt)
       that contain scalar spherical harmonic coefficients
       of the divergence array divg as computed by subroutine shae

av,bv  two or three dimensional arrays (see input parameter nt)
       that contain scalar spherical harmonic coefficients
       of the vorticity array vort as computed by subroutine shaes
       ad,bd,av,bv must be computed by shaes prior to calling idvt

mdab   the first dimension of the arrays ad,bd,av,bv as it appears
       in the program that calls idvtgs (and shags).

ndab   the second dimension of the arrays ad,bd,av,bv as it appear
       the program that calls idvtgs (and shags).

wvhsgs an array which must be initialized by subroutine vhsgsi.

lvhsgs the dimension of the array wvhsgs as it appears in the
       program that calls idvtgs.

work   a work array that does not have to be saved.

lwork  the dimension of the array work as it appears in the
       program that calls idvtgs.

************************************************************
```

```
      output parameters

      v,w   two or three dimensional arrays (see input parameter nt) tha
            contain a vector field whose divergence is divg - pertbd and
            whose vorticity is vort - pertbv.
      pertbd  a nt dimensional array (see input parameter nt and assume
            for the description that follows).  divg - pertbd is a scala
            field which can be the divergence of a vector field (v,w).
        pertbv a nt dimensional array (see input parameter nt and assume
            for the description that follows).  vort - pertbv is a scala
            field which can be the vorticity of a vector field (v,w).

      ierror = 0  no errors
            = 1  error in the specification of nlat
            = 2  error in the specification of nlon
            = 3  error in the specification of isym
            = 4  error in the specification of nt
            = 5  error in the specification of idvw
            = 6  error in the specification of jdvw
            = 7  error in the specification of mdab
            = 8  error in the specification of ndab
            = 9  error in the specification of lvhsgs
            = 10 error in the specification of lwork
      ****************************************************************
```

igradec(...)

```
      ****************************************************************

      let br,bi,cr,ci be the vector spherical harmonic coefficients
      precomputed by vhaec for a vector field (v,w).  let (v',w') be
      the irrotational component of (v,w) (i.e., (v',w') is generated
      by assuming cr,ci are zero and synthesizing br,bi with vhsec).
      then subroutine igradec computes a scalar field sf such that

             gradient(sf) = (v',w').

      i.e.,

             v'(i,j) = d(sf(i,j))/dtheta          (colatitudinal compone
                                                  the gradient)
      and

             w'(i,j) = 1/sint*d(sf(i,j))/dlambda  (east longitudinal com
                                                  of the gradient)

      at colatitude

             theta(i) = (i-1)*pi/(nlat-1)

      and longitude
```

```
       lambda(j) = (j-1)*2*pi/nlon

where sint = sin(theta(i)).  required associated legendre polynomi
are recomputed rather than stored as they are in subroutine igrade
saves storage (compare lshsec and lshses in igrades) but increases
computational requirements.

note:  for an irrotational vector field (v,w), subroutine igradec
computes a scalar field whose gradient is (v,w).  in ay case,
subroutine igradec inverts the gradient subroutine gradec.


************************************************************

input parameters

nlat   the number of colatitudes on the full sphere including the
       poles.

nlon   the number of distinct londitude points.

isym   a parameter which determines whether the scalar field sf is
       computed on the full or half sphere as follows:

 = 0

       the symmetries/antsymmetries described in isym=1,2 below
       do not exist in (v,w) about the equator.  in this case sf
       is neither symmetric nor antisymmetric about the equator.
       sf is computed on the entire sphere.

nt     nt is the number of scalar and vector fields.

isf    the first dimension of the array sf as it appears in
       the program that calls igradec.

jsf    the second dimension of the array sf as it appears in
       the program that calls igradec.

br,bi  two or three dimensional arrays (see input parameter nt)
       that contain vector spherical harmonic coefficients
       of the vector field (v,w) as computed by subroutine vhaec.
       br,bi must be computed by vhaec prior to calling igradec.

mdb    the first dimension of the arrays br and bi as it appears i
       the program that calls igradec (and vhaec).

ndb    the second dimension of the arrays br and bi as it appears
       the program that calls igradec (and vhaec).

wshsec an array which must be initialized by subroutine shseci.
```

```
      lshsec the dimension of the array wshsec as it appears in the
             program that calls igradec.

      work   a work array that does not have to be saved.

      lwork  the dimension of the array work as it appears in the
             program that calls igradec.

      ****************************************************************

      output parameters

      sf    a two or three dimensional array (see input parameter nt) th
            contain a scalar field whose gradient is the irrotational
            component of the vector field (v,w).

      ierror= 0  no errors
            = 1  error in the specification of nlat
            = 2  error in the specification of nlon
            = 3  error in the specification of isym
            = 4  error in the specification of nt
            = 5  error in the specification of isf
            = 6  error in the specification of jsf
            = 7  error in the specification of mdb
            = 8  error in the specification of ndb
            = 9  error in the specification of lshsec
            = 10 error in the specification of lwork
      ****************************************************************
```

igrades(...)

```
      ****************************************************************

      let br,bi,cr,ci be the vector spherical harmonic coefficients
      precomputed by vhaes for a vector field (v,w).  let (v',w') be
      the irrotational component of (v,w) (i.e., (v',w') is generated
      by assuming cr,ci are zero and synthesizing br,bi with vhses).
      then subroutine igrades computes a scalar field sf such that

             gradient(sf) = (v',w').

      i.e.,

             v'(i,j) = d(sf(i,j))/dtheta          (colatitudinal compone
                                                  the gradient)
      and

             w'(i,j) = 1/sint*d(sf(i,j))/dlambda  (east longitudinal com
                                                  of the gradient)

      at colatitude

             theta(i) = (i-1)*pi/(nlat-1)
```

```
and longitude

       lambda(j) = (j-1)*2*pi/nlon
where sint = sin(theta(i)).  required associated legendre polynomi
are stored rather than recomputed as they are in subroutine igrade
note:  for an irrotational vector field (v,w), subroutine igrades
computes a scalar field whose gradient is (v,w).  in ay case,
subroutine igrades inverts the gradient subroutine grades.

************************************************************

input parameters

nlat   the number of colatitudes on the full sphere including the
       poles.
nlon   the number of distinct londitude points.

isym   a parameter which determines whether the scalar field sf is
       computed on the full or half sphere as follows:

 = 0

       the symmetries/antsymmetries described in isym=1,2 below
       do not exist in (v,w) about the equator.

nt     nt is the number of scalar and vector fields.

isf    the first dimension of the array sf as it appears in
       the program that calls igrades.

jsf    the second dimension of the array sf as it appears in
       the program that calls igrades.

br,bi  two or three dimensional arrays (see input parameter nt)
       that contain vector spherical harmonic coefficients
       of the vector field (v,w) as computed by subroutine vhaes.

mdb    the first dimension of the arrays br and bi as it appears i
       the program that calls igrades (and vhaes).

ndb    the second dimension of the arrays br and bi as it appears
       the program that calls igrades (and vhaes).

 wshses an array which must be initialized by subroutine igradesi
       (or equivalently by subroutine shsesi).

lshses the dimension of the array wshses as it appears in the
       program that calls igrades.

work   a work array that does not have to be saved.
```

```
      lwork  the dimension of the array work as it appears in the
             program that calls igrades.

      ************************************************************

      output parameters

      sf     a two or three dimensional array (see input parameter nt) th
             contain a scalar field whose gradient is the irrotational
             component of the vector field (v,w).

      ierror= 0  no errors
            = 1  error in the specification of nlat
            = 2  error in the specification of nlon
            = 3  error in the specification of isym
            = 4  error in the specification of nt
            = 5  error in the specification of isf
            = 6  error in the specification of jsf
            = 7  error in the specification of mdb
            = 8  error in the specification of ndb
            = 9  error in the specification of lshses
            = 10 error in the specification of lwork
      ************************************************************
```

igradgc(...)

```
      ************************************************************

      let br,bi,cr,ci be the vector spherical harmonic coefficients
      precomputed by vhagc for a vector field (v,w).  let (v',w') be
      the irrotational component of (v,w) (i.e., (v',w') is generated
      by assuming cr,ci are zero and synthesizing br,bi with vhsgs).
      then subroutine igradgc computes a scalar field sf such that

             gradient(sf) = (v',w').

      i.e.,

             v'(i,j) = d(sf(i,j))/dtheta          (colatitudinal compone
                                                  the gradient)
      and

             w'(i,j) = 1/sint*d(sf(i,j))/dlambda  (east longitudinal com
                                                  of the gradient)

      at the gaussian colatitude theta(i) (see nlat as input parameter)
      and longitude lambda(j) = (j-1)*2*pi/nlon where sint = sin(theta(i

      note:  for an irrotational vector field (v,w), subroutine igradgc
      computes a scalar field whose gradient is (v,w).  in ay case,
      subroutine igradgc inverts the gradient subroutine gradgc.
```

```
************************************************************

input parameters

nlat   the number of points in the gaussian colatitude grid on the
       full sphere.

nlon   the number of distinct longitude points.

isym   a parameter which determines whether the scalar field sf is
       computed on the full or half sphere as follows:

 = 0

       the symmetries/antsymmetries described in isym=1,2 below
       do not exist in (v,w) about the equator.  in this case sf
       is neither symmetric nor antisymmetric about the equator.
       sf is computed on the entire sphere.

nt     nt is the number of scalar and vector fields.

isf    the first dimension of the array sf as it appears in
       the program that calls igradgc.

jsf    the second dimension of the array sf as it appears in
       the program that calls igradgc.

br,bi  two or three dimensional arrays (see input parameter nt)
       that contain vector spherical harmonic coefficients
       of the vector field (v,w) as computed by subroutine vhagc.
       br,bi must be computed by vhagc prior to calling igradgc.

mdb    the first dimension of the arrays br and bi as it appears i
       the program that calls igradgc (and vhagc).

ndb    the second dimension of the arrays br and bi as it appears
       the program that calls igradgc (and vhagc).

wshsgc    an array which must be initialized by subroutine shsgci.

lshsgc    the dimension of the array wshsgc as it appears in the
       program that calls igradgc.

work   a work array that does not have to be saved.

lwork  the dimension of the array work as it appears in the
       program that calls igradgc

************************************************************

output parameters
```

```
      sf    a two or three dimensional array (see input parameter nt) th
            contain a scalar field whose gradient is the irrotational
            component of the vector field (v,w).

      ierror= 0  no errors
            = 1  error in the specification of nlat
            = 2  error in the specification of nlon
            = 3  error in the specification of isym
            = 4  error in the specification of nt
            = 5  error in the specification of isf
            = 6  error in the specification of jsf
            = 7  error in the specification of mdb
            = 8  error in the specification of ndb
            = 9  error in the specification of lshsgc
            = 10 error in the specification of lwork
      ************************************************************
```

igradgs(...)

```
      ************************************************************

      let br,bi,cr,ci be the vector spherical harmonic coefficients
      precomputed by vhags for a vector field (v,w).  let (v',w') be
      the irrotational component of (v,w) (i.e., (v',w') is generated
      by assuming cr,ci are zero and synthesizing br,bi with vhsgs).
      then subroutine igradgs computes a scalar field sf such that

             gradient(sf) = (v',w').

      i.e.,

             v'(i,j) = d(sf(i,j))/dtheta          (colatitudinal compone
                                                  the gradient)
      and

             w'(i,j) = 1/sint*d(sf(i,j))/dlambda  (east longitudinal com
                                                  of the gradient)

      at the gaussian colatitude theta(i) (see nlat as input parameter)
      and longitude lambda(j) = (j-1)*2*pi/nlon where sint = sin(theta(i

      note:  for an irrotational vector field (v,w), subroutine igradgs
      computes a scalar field whose gradient is (v,w).  in ay case,
      subroutine igradgs inverts the gradient subroutine gradgs.

      ************************************************************

      input parameters

      nlat   the number of points in the gaussian colatitude grid on the
             full sphere.
```

```
nlon   the number of distinct longitude points.

isym   a parameter which determines whether the scalar field sf is
       computed on the full or half sphere as follows:

 = 0

       the symmetries/antsymmetries described in isym=1,2 below
       do not exist in (v,w) about the equator.  in this case sf
       is neither symmetric nor antisymmetric about the equator.
       sf is computed on the entire sphere.

nt     nt is the number of scalar and vector fields.

isf    the first dimension of the array sf as it appears in
       the program that calls igradgs.

jsf    the second dimension of the array sf as it appears in
       the program that calls igradgs.

br,bi  two or three dimensional arrays (see input parameter nt)
       that contain vector spherical harmonic coefficients
       of the vector field (v,w) as computed by subroutine vhags.
       br,bi must be computed by vhags prior to calling igradgs.

mdb    the first dimension of the arrays br and bi as it appears i
       the program that calls igradgs (and vhags).

ndb    the second dimension of the arrays br and bi as it appears
       the program that calls igradgs (and vhags).

wshsgs an array which must be initialized by subroutine igradgsi
       (or equivalently by subroutine shsesi).

lshsgs the dimension of the array wshsgs as it appears in the
       program that calls igradgs.

work   a work array that does not have to be saved.

lwork  the dimension of the array work as it appears in the
       program that calls igradgs.

**************************************************************

output parameters

sf    a two or three dimensional array (see input parameter nt) th
      contain a scalar field whose gradient is the irrotational
      component of the vector field (v,w).

ierror= 0  no errors
      = 1  error in the specification of nlat
```

```
            = 2  error in the specification of nlon
            = 3  error in the specification of isym
            = 4  error in the specification of nt
            = 5  error in the specification of isf
            = 6  error in the specification of jsf
            = 7  error in the specification of mdb
            = 8  error in the specification of ndb
            = 9  error in the specification of lshsgs
            = 10 error in the specification of lwork
      ************************************************************
```

isfvpec(...)

```
      ************************************************************

      given the scalar spherical harmonic coefficients as,bs precomputed
      by shaec for the scalar stream function sf and av,bv precomputed b
      shaec for the scalar velocity potenital vp, subroutine isfvpec com
      the vector field (v,w) corresponding to sf and vp.  w is the east
      longitudinal and v is the colatitudinal component of the vector fi
      (v,w) is expressed in terms of sf,vp by the helmholtz relations (i
      mathematical spherical coordinates):

           v = -1/sin(theta)*d(vp)/dlambda + d(st)/dtheta

           w =  1/sin(theta)*d(st)/dlambda + d(vp)/dtheta

      required legendre functions are recomputed rather than stored as
      they are in subroutine isfvpes.  v(i,j) and w(i,j) are given at
      colatitude

             theta(i) = (i-1)*pi/(nlat-1)

      and east longitude

             lambda(j) = (j-1)*2*pi/nlon

      on the sphere (pi=4.0*atan(1.0)).

      ************************************************************

      input parameters

      nlat   the number of colatitudes on the full sphere including the
             poles.

      nlat   the number of colatitudes on the full sphere including the
      nlon   the number of distinct londitude points.

      isym   a parameter which determines whether the vector field is
             computed on the full or half sphere as follows:

       = 0
```

```
            the symmetries/antsymmetries described in isym=1,2 below
            do not exist in sf,vp about the equator.  in this case v
            and w are not necessarily symmetric or antisymmetric about
            equator.  v and w are computed on the entire sphere.

     nt     nt is the number of scalar and vector fields.

     idv    the first dimension of the arrays v,w as it appears in
            the program that calls isfvpec.

     jdv    the second dimension of the arrays v,w as it appears in
            the program that calls isfvpec.

     as,bs  two or three dimensional arrays (see input parameter nt)
            that contain the spherical harmonic coefficients of
            the scalar field sf as computed by subroutine shaec.

     av,bv  two or three dimensional arrays (see input parameter nt)
            that contain the spherical harmonic coefficients of
            the scalar field vp as computed by subroutine shaec.

     mdb    the first dimension of the arrays as,bs,av,bv as it
            appears in the program that calls isfvpec.

     ndb    the second dimension of the arrays as,bs,av,bv as it
            appears in the program that calls isfvpec.

     wvhsec an array which must be initialized by subroutine vhseci.

     lvhsec the dimension of the array wvhsec as it appears in the
            program that calls isfvpec.

     work   a work array that does not have to be saved.

     lwork  the dimension of the array work as it appears in the
            program that calls isfvpec.

     ****************************************************************

     output parameters

     v,w    two or three dimensional arrays (see input parameter nt)
           that contains the vector field corresponding to the stream
           function sf and velocity potential vp whose coefficients,
           as,bs (for sf) and av,bv (for vp), were precomputed by
           subroutine shaec.
           colatitude point

     ierror = 0  no errors
           = 1  error in the specification of nlat
           = 2  error in the specification of nlon
```

```
           = 3  error in the specification of isym
           = 4  error in the specification of nt
           = 5  error in the specification of idv
           = 6  error in the specification of jdv
           = 7  error in the specification of mdb
           = 8  error in the specification of ndb
           = 9  error in the specification of lvhsec
           = 10 error in the specification of lwork
     ************************************************************
```

isfvpes(...)

```
     ************************************************************

       given the scalar spherical harmonic coefficients as,bs precomput
       by shaes for the scalar stream function sf and av,bv precomputed
       shaes for the scalar velocity potenital vp, subroutine isfvpes c
       the vector field (v,w) corresponding to sf and vp.  w is the eas
       longitudinal and v is the colatitudinal component of the vector
       (v,w) is expressed in terms of sf,vp by the helmholtz relations
       mathematical spherical coordinates):

            v = -1/sin(theta)*d(vp)/dlambda + d(st)/dtheta

            w =  1/sin(theta)*d(st)/dlambda + d(vp)/dtheta

       required legendre functions are stored rather than recomputed as
       they are in subroutine isfvpes.  v(i,j) and w(i,j) are given at
       colatitude

              theta(i) = (i-1)*pi/(nlat-1)

       and east longitude

              lambda(j) = (j-1)*2*pi/nlon

       on the sphere (pi=4.0*atan(1.0)).


     ************************************************************

       input parameters

       nlat   the number of colatitudes on the full sphere including th
              poles.

       nlon   the number of distinct longitude points.

       isym   a parameter which determines whether the vector field is
              computed on the full or half sphere as follows:

        = 0
              the symmetries/antsymmetries described in isym=1,2 below
```

```
            do not exist in sf,vp about the equator. v and w are
            computed on the entire sphere.

     nt     nt is the number of scalar and vector fields.

     idv    the first dimension of the arrays v,w as it appears in
            the program that calls isfvpes.

     jdv    the second dimension of the arrays v,w as it appears in
            the program that calls isfvpes. jdv must be at least nlon

     as,bs  two or three dimensional arrays (see input parameter nt)
            that contain the spherical harmonic coefficients of
            the scalar field sf as computed by subroutine shaes.

     av,bv  two or three dimensional arrays (see input parameter nt)
            that contain the spherical harmonic coefficients of
            the scalar field vp as computed by subroutine shaes.

     mdb    the first dimension of the arrays as,bs,av,bv as it
            appears in the program that calls isfvpes.

     ndb    the second dimension of the arrays as,bs,av,bv as it
            appears in the program that calls isfvpes.

     wvhses an array which must be initialized by subroutine vhsesi.

     lvhses the dimension of the array wvhses as it appears in the
            program that calls isfvpes.

     work   a work array that does not have to be saved.

     lwork  the dimension of the array work as it appears in the
            program that calls isfvpes.

   ************************************************************

     output parameters

    v,w     two or three dimensional arrays (see input parameter nt)
            that contains the vector field corresponding to the stream
            function sf and velocity potential vp whose coefficients,
            as,bs (for sf) and av,bv (for vp), were precomputed by
            subroutine shaes.

    ierror = 0  no errors
           = 1  error in the specification of nlat
           = 2  error in the specification of nlon
           = 3  error in the specification of isym
           = 4  error in the specification of nt
           = 5  error in the specification of idv
           = 6  error in the specification of jdv
```

```
            = 7  error in the specification of mdb
            = 8  error in the specification of ndb
            = 9  error in the specification of lvhses
            = 10 error in the specification of lwork
     ****************************************************************
```

isfvpgc(...)

```
     ****************************************************************

     given the scalar spherical harmonic coefficients as,bs precomputed
     by shagc for the scalar stream function sf and av,bv precomputed b
     shagc for the scalar velocity potenital vp, subroutine isfvpgc com
     the vector field (v,w) corresponding to sf and vp.  w is the east
     longitudinal and v is the colatitudinal component of the vector fi
     (v,w) is expressed in terms of sf,vp by the helmholtz relations (i
     mathematical spherical coordinates):

          v = -1/sin(theta)*d(vp)/dlambda + d(st)/dtheta

          w =  1/sin(theta)*d(st)/dlambda + d(vp)/dtheta

     required legendre functions are recomputed rather than stored as
     they are in subroutine isfvpgs.  v(i,j) and w(i,j) are given at
     the i(th) gaussian colatitude point (see gaqd) theta(i) and east
     longitude lambda(j) = (j-1)*2.*pi/nlon on the sphere.


     ************************************************************

     input parameters

     nlat   the number of points in the gaussian colatitude grid on the
            full sphere.

     nlon   the number of distinct longitude points.

     isym   a parameter which determines whether the vector field is
            computed on the full or half sphere as follows:

      = 0

            the symmetries/antsymmetries described in isym=1,2 below
            do not exist in sf,vp about the equator.  in this case v
            and w are not necessarily symmetric or antisymmetric about
            equator.  v and w are computed on the entire sphere.

     nt     nt is the number of scalar and vector fields.

     idv    the first dimension of the arrays v,w as it appears in
            the program that calls isfvpgc.

     jdv    the second dimension of the arrays v,w as it appears in
```

```
       the program that calls isfvpgc.

as,bs  two or three dimensional arrays (see input parameter nt)
       that contain the spherical harmonic coefficients of
       the scalar field sf as computed by subroutine shagc.

av,bv  two or three dimensional arrays (see input parameter nt)
       that contain the spherical harmonic coefficients of
       the scalar field vp as computed by subroutine shagc.

mdb    the first dimension of the arrays as,bs,av,bv as it
       appears in the program that calls isfvpgc.

ndb    the second dimension of the arrays as,bs,av,bv as it
       appears in the program that calls isfvpgc.

wvhsgc an array which must be initialized by subroutine vhsgci.

lvhsgc the dimension of the array wvhsgc as it appears in the
       program that calls isfvpgc.

work   a work array that does not have to be saved.

lwork  the dimension of the array work as it appears in the

       program that calls isfvpgc.

**************************************************************

 output parameters



v,w    two or three dimensional arrays (see input parameter nt)
      that contains the vector field corresponding to the stream
      function sf and velocity potential vp whose coefficients,
      as,bs (for sf) and av,bv (for vp), were precomputed by
      subroutine shagc.

ierror= 0  no errors
      = 1  error in the specification of nlat
      = 2  error in the specification of nlon
      = 3  error in the specification of isym
      = 4  error in the specification of nt
      = 5  error in the specification of idv
      = 6  error in the specification of jdv
      = 7  error in the specification of mdb
      = 8  error in the specification of ndb
      = 9  error in the specification of lvhsgc
      = 10 error in the specification of lwork
**************************************************************
```

```
     ************************************************************

     given the scalar spherical harmonic coefficients as,bs precomputed
     by shags for the scalar stream function sf and av,bv precomputed b
     shags for the scalar velocity potenital vp, subroutine isfvpgs com
     the vector field (v,w) corresponding to sf and vp.  w is the east
     longitudinal and v is the colatitudinal component of the vector fi
     (v,w) is expressed in terms of sf,vp by the helmholtz relations (i
     mathematical spherical coordinates):

          v = -1/sin(theta)*d(vp)/dlambda + d(st)/dtheta

          w =  1/sin(theta)*d(st)/dlambda + d(vp)/dtheta

     required legendre functions are stored rather than recomputed as
     they are in subroutine isfvpgc.  v(i,j) and w(i,j) are given at
     the i(th) gaussian colatitude point (see gaqd) theta(i) and east
     longitude lambda(j) = (j-1)*2.*pi/nlon on the sphere.


     ************************************************************

     input parameters

     nlat   the number of points in the gaussian colatitude grid on the
            full sphere.

     nlon   the number of distinct londitude points.

     isym   a parameter which determines whether the vector field is
            computed on the full or half sphere as follows:

      = 0

            the symmetries/antsymmetries described in isym=1,2 below
            do not exist in sf,vp about the equator.  in this case v
            and w are not necessarily symmetric or antisymmetric about
            equator.  v and w are computed on the entire sphere.

     nt     nt is the number of scalar and vector fields.

     idv    the first dimension of the arrays v,w as it appears in
            the program that calls isfvpgs.

     jdv    the second dimension of the arrays v,w as it appears in
            the program that calls isfvpgs.

     as,bs  two or three dimensional arrays (see input parameter nt)
            that contain the spherical harmonic coefficients of
            the scalar field sf as computed by subroutine shags.
```

```
      av,bv  two or three dimensional arrays (see input parameter nt)
             that contain the spherical harmonic coefficients of
             the scalar field vp as computed by subroutine shags.

      mdb    the first dimension of the arrays as,bs,av,bv as it
             appears in the program that calls isfvpgs.

      ndb    the second dimension of the arrays as,bs,av,bv as it
             appears in the program that calls isfvpgs.

      wvhsgs an array which must be initialized by subroutine vhsgsi.

      lvhsgs the dimension of the array wvhsgs as it appears in the
             program that calls isfvpgs.

      lwork  the dimension of the array work as it appears in the
             program that calls isfvpgs.

      ****************************************************************

      output parameters

      v,w    two or three dimensional arrays (see input parameter nt)
            that contains the vector field corresponding to the stream
            function sf and velocity potential vp whose coefficients,
            as,bs (for sf) and av,bv (for vp), were precomputed by
            subroutine shags.

      ierror = 0  no errors
            = 1  error in the specification of nlat
            = 2  error in the specification of nlon
            = 3  error in the specification of isym
            = 4  error in the specification of nt
            = 5  error in the specification of idv
            = 6  error in the specification of jdv
            = 7  error in the specification of mdb
            = 8  error in the specification of ndb
            = 9  error in the specification of lvhsgs
            = 10 error in the specification of lwork
      ****************************************************************
```

islapec(...)

```
      ****************************************************************

      islapec inverts the laplace or helmholz operator on an equally
      spaced latitudinal grid using o(n**2) storage. given the
      spherical harmonic coefficients a(m,n) and b(m,n) of the right
      hand side slap(i,j), islapec computes a solution sf(i,j) to
      the following helmhotz equation :

            2                  2
      [d(sf(i,j))/dlambda /sint + d(sint*d(sf(i,j))/dtheta)/dtheta]/sint
```

```
              - xlmbda * sf(i,j) = slap(i,j)

 where sf(i,j) is computed at colatitude

            theta(i) = (i-1)*pi/(nlat-1)

       and longitude

            lambda(j) = (j-1)*2*pi/nlon

       for i=1,...,nlat and j=1,...,nlon.


************************************************************


input parameters

nlat   the number of colatitudes on the full sphere including the
       poles.
nlon   the number of distinct longitude points.

isym   this parameter should have the same value input to subrouti
       shaec to compute the coefficients a and b for the scalar fi
       slap.  isym is set as follows:

       = 0  no symmetries exist in slap about the equator. scalar
            synthesis is used to compute sf on the entire sphere.

nt     the number of solutions.

xlmbda  a one dimensional array with nt elements. if xlmbda is
       is identically zero islapec solves poisson's equation.
       if xlmbda > 0.0 islapec solves the helmholtz equation.

ids    the first dimension of the array sf as it appears in the
       program that calls islapec.

jds    the second dimension of the array sf as it appears in the
       program that calls islapec.


a,b    two or three dimensional arrays (see input parameter nt)
       that contain scalar spherical harmonic coefficients
       of the scalar field slap. a,b must be computed by shaec
       prior to calling islapec.


mdab   the first dimension of the arrays a and b as it appears
       in the program that calls islapec.
```

```
      ndab   the second dimension of the arrays a and b as it appears
             in the program that calls islapec.

             mdab,ndab should have the same values input to shaec to
             compute the coefficients a and b.

      wshsec an array which must be initialized by subroutine shseci.

      lshsec the dimension of the array wshsec as it appears in the
             program that calls islapec.

      work   a work array that does not have to be saved.

      lwork    the dimension of the array work as it appears in the
             program that calls islapec.

      ****************************************************************

      output parameters

      sf      two or three dimensional arrays (see input parameter nt)
             that contain the solution to either the helmholtz
             (xlmbda>0.0) or poisson's equation.

      pertrb a one dimensional array with nt elements

      ierror   a parameter which flags errors in input parameters as fol
             =-1  xlmbda is input negative (nonfatal error)
             = 0  no errors detected
             = 1  error in the specification of nlat
             = 2  error in the specification of nlon
             = 3  error in the specification of ityp
             = 4  error in the specification of nt
             = 5  error in the specification of ids
             = 6  error in the specification of jds
             = 7  error in the specification of mdbc
             = 8  error in the specification of ndbc
             = 9  error in the specification of lsave
             = 10 error in the specification of lwork
      ****************************************************************
```

islapes(...)

```
      ****************************************************************

      islapes inverts the laplace or helmholz operator on an equally
      spaced latitudinal grid using o(n**3) storage. given the
      spherical harmonic coefficients a(m,n) and b(m,n) of the right
      hand side slap(i,j), islapes computes a solution sf(i,j) to
      the following helmhotz equation :

            2                  2
      [d(sf(i,j))/dlambda /sint + d(sint*d(sf(i,j))/dtheta)/dtheta]/sint
```

```
              - xlmbda * sf(i,j) = slap(i,j)

 where sf(i,j) is computed at colatitude

            theta(i) = (i-1)*pi/(nlat-1)

       and longitude

            lambda(j) = (j-1)*2*pi/nlon

       for i=1,...,nlat and j=1,...,nlon.

************************************************************

input parameters

nlat   the number of colatitudes on the full sphere including the
       poles.

nlon   the number of distinct longitude points.

isym   this parameter should have the same value input to subrouti
       shaes to compute the coefficients a and b for the scalar fi
       slap.  isym is set as follows:

       = 0  no symmetries exist in slap about the equator. scalar
            synthesis is used to compute sf on the entire sphere.

nt     the number of solutions.

xlmbda a one dimensional array with nt elements. if xlmbda is
       is identically zero islapes solves poisson's equation.
       if xlmbda > 0.0 islapes solves the helmholtz equation.

ids    the first dimension of the array sf as it appears in the
       program that calls islapes.

jds    the second dimension of the array sf as it appears in the
       program that calls islapes.


a,b    two or three dimensional arrays (see input parameter nt)
       that contain scalar spherical harmonic coefficients
       of the scalar field slap. a,b must be computed by shaes
       prior to calling islapes.


mdab   the first dimension of the arrays a and b as it appears
       in the program that calls islapes.

ndab   the second dimension of the arrays a and b as it appears
```

```
            in the program that calls islapes.

            mdab,ndab should have the same values input to shaes to
            compute the coefficients a and b.


      wshses an array which must be initialized by subroutine shsesi.

      lshses the dimension of the array wshses as it appears in the
            program that calls islapes.

      work   a work array that does not have to be saved.

      lwork  the dimension of the array work as it appears in the
            program that calls islapes.

      ****************************************************************

      output parameters


      sf      a two or three dimensional arrays (see input parameter nt)
            inverts the scalar laplacian in slap - pertrb.

      pertrb  a one dimensional array with nt elements.

      ierror    a parameter which flags errors in input parameters as fo
            = 0  no errors detected
            = 1  error in the specification of nlat
            = 2  error in the specification of nlon
            = 3  error in the specification of ityp
            = 4  error in the specification of nt
            = 5  error in the specification of ids
            = 6  error in the specification of jds
            = 7  error in the specification of mdbc
            = 8  error in the specification of ndbc
            = 9  error in the specification of lshses
            = 10 error in the specification of lwork
      ****************************************************************
```

islapgc(...)

```
      ****************************************************************

      islapgc inverts the laplace or helmholz operator on a Gaussian
      grid using o(n**2) storage. given the
      spherical harmonic coefficients a(m,n) and b(m,n) of the right
      hand side slap(i,j), islapgc computes a solution sf(i,j) to
      the following helmhotz equation :

            2                  2
      [d(sf(i,j))/dlambda /sint + d(sint*d(sf(i,j))/dtheta)/dtheta]/sint
```

```
              - xlmbda * sf(i,j) = slap(i,j)

 where sf(i,j) is computed at the Gaussian colatitude point theta(
 (see nlat as an input argument) and longitude

            lambda(j) = (j-1)*2*pi/nlon

       for i=1,...,nlat and j=1,...,nlon.

************************************************************

input parameters

nlat   the number of points in the gaussian colatitude grid on the
       full sphere.

nlon   the number of distinct longitude points.

isym   this parameter should have the same value input to subrouti
       shagc to compute the coefficients a and b for the scalar fi
       slap.  isym is set as follows:

       = 0  no symmetries exist in slap about the equator. scalar
            synthesis is used to compute sf on the entire sphere.
            i.e., in the array sf(i,j) for i=1,...,nlat and
            j=1,...,nlon.

nt     the number of solutions.

xlmbda a one dimensional array with nt elements. if xlmbda is
       is identically zero islapgc solves poisson's equation.
       if xlmbda > 0.0 islapgc solves the helmholtz equation.

ids    the first dimension of the array sf as it appears in the
       program that calls islapgc.

jds    the second dimension of the array sf as it appears in the
       program that calls islapgc.


a,b    two or three dimensional arrays (see input parameter nt)
       that contain scalar spherical harmonic coefficients
       of the scalar field slap. a,b must be computed by shagc
       prior to calling islapgc.


mdab   the first dimension of the arrays a and b as it appears
       in the program that calls islapgc.

ndab   the second dimension of the arrays a and b as it appears
       in the program that calls islapgc.
```

```
       mdab,ndab should have the same values input to shagc to
       compute the coefficients a and b.

wshsgc an array which must be initialized by subroutine shsgci
       once initialized, wshsgc can be used repeatedly by islapgc
       as long as nlon and nlat remain unchanged.  wshsgc must
       not be altered between calls of islapgc.

lshsgc the dimension of the array wshsgc as it appears in the
       program that calls islapgc.

work   a work array that does not have to be saved.

lwork  the dimension of the array work as it appears in the
       program that calls islapgc.

*************************************************************

output parameters

sf     a two or three dimensional arrays (see input parameter nt)
       inverts the scalar laplacian in slap.

pertrb a one dimensional array with nt elements.

ierror    a parameter which flags errors in input parameters as fo
       = 0  no errors detected
       = 1  error in the specification of nlat
       = 2  error in the specification of nlon
       = 3  error in the specification of ityp
       = 4  error in the specification of nt
       = 5  error in the specification of ids
       = 6  error in the specification of jds
       = 7  error in the specification of mdbc
       = 8  error in the specification of ndbc
       = 9  error in the specification of lshsgc
       = 10 error in the specification of lwork
*************************************************************
```

islapgs(...)

```
*************************************************************

islapgs inverts the laplace or helmholz operator on a Gaussian gri
Given the spherical harmonic coefficients a(m,n) and b(m,n) of the
right hand side slap(i,j), islapgc computes a solution sf(i,j) to
the following helmhotz equation :

      2                 2
[d(sf(i,j))/dlambda /sint + d(sint*d(sf(i,j))/dtheta)/dtheta]/sint

              - xlmbda * sf(i,j) = slap(i,j)
```

```
 where sf(i,j) is computed at the Gaussian colatitude point theta(
 (see nlat as an input argument) and longitude

            lambda(j) = (j-1)*2*pi/nlon

       for i=1,...,nlat and j=1,...,nlon.

************************************************************

input parameters

nlat   the number of points in the gaussian colatitude grid on the
       full sphere.

nlon   the number of distinct longitude points.

isym   this parameter should have the same value input to subrouti
       shags to compute the coefficients a and b for the scalar fi
       slap.  isym is set as follows:

       = 0  no symmetries exist in slap about the equator. scalar
            synthesis is used to compute sf on the entire sphere.

nt     the number of analyses.

xlmbda   a one dimensional array with nt elements. if xlmbda is
       is identically zero islapgc solves poisson's equation.
       if xlmbda > 0.0 islapgc solves the helmholtz equation.

ids      the first dimension of the array sf as it appears in the
       program that calls islapgs.

jds      the second dimension of the array sf as it appears in the
       program that calls islapgs.

a,b      two or three dimensional arrays (see input parameter nt)
       that contain scalar spherical harmonic coefficients
       of the scalar field slap as computed by subroutine shags.
       a,b must be computed by shags prior to calling islapgs.

mdab    the first dimension of the arrays a and b as it appears
       in the program that calls islapgs.

ndab    the second dimension of the arrays a and b as it appears
       in the program that calls islapgs.

wshsgs  an array which must be initialized by subroutine islapgsi
       (or equivalently by shsesi).

lshsgs  the dimension of the array wshsgs as it appears in the
       program that calls islapgs.
```

```
      work   a work array that does not have to be saved.

      lwork  the dimension of the array work as it appears in the
             program that calls islapgs.

      ****************************************************************

      output parameters

      sf      a two or three dimensional arrays (see input parameter nt)
             inverts the scalar laplacian in slap.

      pertrb  a one dimensional array with nt elements.

      ierror   a parameter which flags errors in input parameters as fol
             = 0  no errors detected
             = 1  error in the specification of nlat
             = 2  error in the specification of nlon
             = 3  error in the specification of ityp
             = 4  error in the specification of nt
             = 5  error in the specification of ids
             = 6  error in the specification of jds
             = 7  error in the specification of mdbc
             = 8  error in the specification of ndbc
             = 9  error in the specification of lshsgs
             = 10 error in the specification of lwork
      ****************************************************************
```

ivlapec(...)

```
      ****************************************************************

      subroutine ivlapec computes a the vector field (v,w) whose vector
      laplacian is (vlap,wlap).  w and wlap are east longitudinal
      components of the vectors.  v and vlap are colatitudinal component
      of the vectors.  br,bi,cr, and ci are the vector harmonic coeffici
      of (vlap,wlap).  these must be precomputed by vhaec and are input
      parameters to ivlapec.  (v,w) have the same symmetry or lack of
      symmetry about the about the equator as (vlap,wlap).  the input
      parameters ityp,nt,mdbc,ndbc must have the same values used by
      vhaec to compute br,bi,cr, and ci for (vlap,wlap).

      ****************************************************************

      input parameters

      nlat   the number of colatitudes on the full sphere including the
             poles.

      nlon   the number of distinct longitude points.

      ityp   this parameter should have the same value input to subrouti
             vhaec to compute the coefficients br,bi,cr, and ci for the
```

```
       vector field (vlap,wlap).  ityp is set as follows:

       = 0  no symmetries exist in (vlap,wlap) about the equator.
            is computed and stored on the entire sphere.

nt     nt is the number of vector fields (vlap,wlap).

idvw     the first dimension of the arrays w and v as it appears i
       the program that calls ivlapec.

jdvw     the second dimension of the arrays w and v as it appears
       the program that calls ivlapec.


br,bi    two or three dimensional arrays (see input parameter nt)
cr,ci    that contain vector spherical harmonic coefficients of th
       vector field (vlap,wlap) as computed by subroutine vhaec.
       br,bi,cr and ci must be computed by vhaec prior to calling
       ivlapec.

mdb           !  the first dimension of the arrays br,bi,cr and ci
       appears in the program that calls ivlapec.

ndb           !  the second dimension of the arrays br,bi,cr and c
       appears in the program that calls ivlapec.

wvhse           !an array which must be initialized by subroutine

lvhse           !the dimension of the array wvhsec as it appears i
       program that calls ivlapec.

work   a work array that does not have to be saved.

lwork  the dimension of the array work as it appears in the
       program that calls ivlapec.

************************************************************

output parameters

v,w     two or three dimensional arrays (see input parameter nt) t
       contain a vector field whose vector laplacian is (vlap,wlap

ierror    a parameter which flags errors in input parameters as fo
       = 0  no errors detected
       = 1  error in the specification of nlat
       = 2  error in the specification of nlon
       = 3  error in the specification of ityp
       = 4  error in the specification of nt
       = 5  error in the specification of idvw
       = 6  error in the specification of jdvw
       = 7  error in the specification of mdbc
```

```
            = 8  error in the specification of ndbc
            = 9  error in the specification of lvhsec
            = 10 error in the specification of lwork
      ************************************************************
```

ivlapes(...)

```
      ************************************************************

        subroutine ivlapes computes a the vector field (v,w) whose vecto
        laplacian is (vlap,wlap).  w and wlap are east longitudinal
        components of the vectors.  v and vlap are colatitudinal compone
        of the vectors.  br,bi,cr, and ci are the vector harmonic coeffi
        of (vlap,wlap).  these must be precomputed by vhaes and are inpu
        parameters to ivlapes.  (v,w) have the same symmetry or lack of
        symmetry about the about the equator as (vlap,wlap).  the input
        parameters ityp,nt,mdbc,ndbc must have the same values used by
        vhaes to compute br,bi,cr, and ci for (vlap,wlap).

      ************************************************************

        input parameters

        nlat   the number of colatitudes on the full sphere including th
               poles.

        nlon   the number of distinct longitude points.

        ityp   this parameter should have the same value input to subrou
               vhaes to compute the coefficients br,bi,cr, and ci for th
               vector field (vlap,wlap).  ityp is set as follows:

               = 0  no symmetries exist in (vlap,wlap) about the equator
                    is computed and stored on the entire sphere.

      idvw     the first dimension of the arrays w and v as it appears i
               the program that calls ivlapes.

      jdvw     the second dimension of the arrays w and v as it appears
               the program that calls ivlapes.

      br,bi    two or three dimensional arrays (see input parameter nt)
      cr,ci    that contain vector spherical harmonic coefficients of th
               vector field (vlap,wlap) as computed by subroutine vhaes.

       mdbc    the first dimension of the arrays br,bi,cr and ci as it
               appears in the program that calls ivlapes.

       ndbc    the second dimension of the arrays br,bi,cr and ci as it
               appears in the program that calls ivlapes.

       wvhses  an array which must be initialized by subroutine vhsesi.
```

```
      lvhses  the dimension of the array wvhses as it appears in the
              program that calls ivlapes.

       work   a work array that does not have to be saved.

       lwork  the dimension of the array work as it appears in the
              program that calls ivlapes.

     ************************************************************

       output parameters

      v,w     two or three dimensional arrays (see input parameter nt)
              contain a vector field whose vector laplacian is (vlap,wl

      ierror  a parameter which flags errors in input parameters as fol
              = 0  no errors detected
              = 1  error in the specification of nlat
              = 2  error in the specification of nlon
              = 3  error in the specification of ityp
              = 4  error in the specification of nt
              = 5  error in the specification of idvw
              = 6  error in the specification of jdvw
              = 7  error in the specification of mdbc
              = 8  error in the specification of ndbc
              = 9  error in the specification of lvhses
              = 10 error in the specification of lwork
     ************************************************************
```

ivlapgc(...)

```
     ************************************************************

       given the vector spherical harmonic coefficients (br,bi,cr,ci)
       precomputed by subroutine vhagc for a vector field (vlap,wlap),
       subroutine ivlapgc computes a vector field (v,w) whose vector
       laplacian is (vlap,wlap).  v,vlap are the colatitudinal
       components and w,wlap are the east longitudinal components of
       the vectors.  (v,w) have the same symmetry or lack of symmetry
       about the equator as (vlap,wlap).  the input parameters ityp,
       nt,mdbc,ndbc must have the same values used by vhagc to compute
       br,bi,cr,ci for (vlap,wlap).


     ************************************************************

       input parameters

       nlat   the number of points in the gaussian colatitude grid on t
              full sphere.

       nlon   the number of distinct longitude points.
```

```
  ityp   this parameter should have the same value input to subrou
         vhagc to compute the coefficients br,bi,cr, and ci for th
         vector field (vlap,wlap).  ityp is set as follows:

         = 0  no symmetries exist in (vlap,wlap) about the equator
              is computed and stored on the entire sphere.

   nt    nt is the number of vector fields (vlap,wlap).

  idvw   the first dimension of the arrays w and v as it appears i
         the program that calls ivlapgc.

  jdvw   the second dimension of the arrays w and v as it appears
         the program that calls ivlapgc.

 br,bi   two or three dimensional arrays (see input parameter nt)
 cr,ci   that contain vector spherical harmonic coefficients of th
         vector field (vlap,wlap) as computed by subroutine vhagc.
         br,bi,cr and ci must be computed by vhagc prior to callin
         ivlapgc.

  mdbc   the first dimension of the arrays br,bi,cr and ci as it
         appears in the program that calls ivlapgc.

  ndbc   the second dimension of the arrays br,bi,cr and ci as it
         appears in the program that calls ivlapgc.

  wvhsgc an array which must be initialized by subroutine vhsgci.

  lvhsgc the dimension of the array wvhsgc as it appears in the
         program that calls ivlapgc.

   work  a work array that does not have to be saved.

   lwork the dimension of the array work as it appears in the
         program that calls ivlapgc.

**************************************************************

   output parameters

  v,w    two or three dimensional arrays (see input parameter nt)
         contain a vector field whose vector laplacian is (vlap,wl

  ierror a parameter which flags errors in input parameters as fol
         = 0  no errors detected
         = 1  error in the specification of nlat
         = 2  error in the specification of nlon
         = 3  error in the specification of ityp
         = 4  error in the specification of nt
         = 5  error in the specification of idvw
         = 6  error in the specification of jdvw
```

```
            = 7  error in the specification of mdbc
            = 8  error in the specification of ndbc
            = 9  error in the specification of lvhsgc
            = 10 error in the specification of lwork
     ****************************************************************
```

ivlapgs(...)

```
     ****************************************************************

       given the vector spherical harmonic coefficients (br,bi,cr,ci)
       precomputed by subroutine vhags for a vector field (vlap,wlap),
       subroutine ivlapgs computes a vector field (v,w) whose vector
       laplacian is (vlap,wlap).  v,vlap are the colatitudinal
       components and w,wlap are the east longitudinal components of
       the vectors.  (v,w) have the same symmetry or lack of symmetry
       about the equator as (vlap,wlap).  the input parameters ityp,
       nt,mdbc,ndbc must have the same values used by vhags to compute
       br,bi,cr,ci for (vlap,wlap).

     ************************************************************

       input parameters

       nlat   the number of points in the gaussian colatitude grid on t
              full sphere.

       nlon   the number of distinct longitude points.

       ityp   this parameter should have the same value input to subrou
              vhags to compute the coefficients br,bi,cr, and ci for th
              vector field (vlap,wlap).

              = 0  no symmetries exist in (vlap,wlap) about the equator
                   is computed and stored on the entire sphere.

       nt     nt is the number of vector fields (vlap,wlap).

     idvw     the first dimension of the arrays w and v as it appears i
              the program that calls ivlapgs.

     jdvw     the second dimension of the arrays w and v as it appears
              the program that calls ivlapgs.

     br,bi    two or three dimensional arrays (see input parameter nt)
     cr,ci    that contain vector spherical harmonic coefficients of th
              vector field (vlap,wlap) as computed by subroutine vhags.
              br,bi,cr and ci must be computed by vhags prior to callin
              ivlapgs.

      mdbc    the first dimension of the arrays br,bi,cr and ci as it
              appears in the program that calls ivlapgs.
```

```
      ndbc    the second dimension of the arrays br,bi,cr and ci as it
              appears in the program that calls ivlapgs.

      wvhsgs  an array which must be initialized by subroutine vhsgsi.

      lvhsgs  the dimension of the array wvhsgs as it appears in the
              program that calls ivlapgs.

       work   a work array that does not have to be saved.

       lwork  the dimension of the array work as it appears in the
              program that calls ivlapgs.

     ****************************************************************

       output parameters

      v,w     two or three dimensional arrays (see input parameter nt)
              contain a vector field whose vector laplacian is (vlap,wl

      ierror  a parameter which flags errors in input parameters as fol
              = 0  no errors detected
              = 1  error in the specification of nlat
              = 2  error in the specification of nlon
              = 3  error in the specification of ityp
              = 4  error in the specification of nt
              = 5  error in the specification of idvw
              = 6  error in the specification of jdvw
              = 7  error in the specification of mdbc
              = 8  error in the specification of ndbc
              = 9  error in the specification of lvhsgs
              = 10 error in the specification of lwork
     ****************************************************************
```

ivrtec(...)

```
     ****************************************************************

       given the scalar spherical harmonic coefficients a and b, precom
       by subroutine shaec for a scalar array vort, subroutine ivrtec c
       a divergence free vector field (v,w) whose vorticity is vt - per
       w is the east longitude component and v is the colatitudinal com
       pertrb is a constant which must be subtracted from vort for (v,w
       exist (see the description of pertrb below).  usually pertrb is
       or small relative to vort.  the divergence of (v,w), as computed
       ivrtec, is the zero scalar field.  i.e., v(i,j) and w(i,j) are t
       colaatitudinal and east longitude velocity components at colatit

              theta(i) = (i-1)*pi/(nlat-1)

       and longitude

              lambda(j) = (j-1)*2*pi/nlon.
```

```
  the

         vorticity(v(i,j),w(i,j))

      =  [-dv/dlambda + d(sint*w)/dtheta]/sint

      =  vort(i,j) - pertrb

  and

         divergence(v(i,j),w(i,j))

      =  [d(sint*v)/dtheta + dw/dlambda]/sint

      =  0.0

  where sint = sin(theta(i)).  required associated legendre polync
  are recomputed rather than stored as they are in subroutine ivrt

*************************************************************


  input parameters

  nlat   the number of colatitudes on the full sphere including th
         poles.

  nlon   the number of distinct longitude points.

  isym   this has the same value as the isym that was input to
         subroutine shaec to compute the arrays a and b.  isym
         determines whether (v,w) are computed on the full or half
         sphere as follows:

   = 0
         vort is not symmetric about the equator. in this case
         the vector field (v,w) is computed on the entire sphere.

  nt     in the program that calls ivrtec, nt is the number of vor
         and vector fields.

  idvw   the first dimension of the arrays v,w as it appears in
         the program that calls ivrtec.

  jdvw   the second dimension of the arrays v,w as it appears in
         the program that calls ivrtec.

  a,b    two or three dimensional arrays (see input parameter nt)
         that contain scalar spherical harmonic coefficients
         of the vorticity array vort as computed by subroutine sha
         a,b must be computed by shaec prior to calling ivrtec.
```

```
  mdab   the first dimension of the arrays a and b as it appears i
         the program that calls ivrtec (and shaec).

  ndab   the second dimension of the arrays a and b as it appears
         the program that calls ivrtec (and shaec). ndab must be a
         least nlat.


 wvhsec  an array which must be initialized by subroutine vhseci.

 lvhsec  the dimension of the array wvhsec as it appears in the
         program that calls ivrtec.

  work   a work array that does not have to be saved.

  lwork  the dimension of the array work as it appears in the
         program that calls ivrtec.

 ************************************************************

 output parameters

 v,w   two or three dimensional arrays (see input parameter nt) th
       contain a divergence free vector field whose vorticity is
       vort - pertrb.

pertrb  a nt dimensional array.

ierror = 0  no errors
       = 1  error in the specification of nlat
       = 2  error in the specification of nlon
       = 3  error in the specification of isym
       = 4  error in the specification of nt
       = 5  error in the specification of idvw
       = 6  error in the specification of jdvw
       = 7  error in the specification of mdab
       = 8  error in the specification of ndab
       = 9  error in the specification of lvhsec
       = 10 error in the specification of lwork
 ************************************************************
```

ivrtes(...)

```
 ************************************************************

  given the scalar spherical harmonic coefficients a and b, precom
  by subroutine shaes for a scalar array vort, subroutine ivrtes c
  a divergence free vector field (v,w) whose vorticity is vort - p
  w is the east longitude component and v is the colatitudinal com
  pertrb is a constant which must be subtracted from vort for (v,w
  exist (see the description of pertrb below).  usually pertrb is
  or small relative to vort.  the divergence of (v,w), as computed
```

```
  ivrtes, is the zero scalar field.  i.e., v(i,j) and w(i,j) are t
  colaatitudinal and east longitude velocity components at colatit

         theta(i) = (i-1)*pi/(nlat-1)

  and longitude

         lambda(j) = (j-1)*2*pi/nlon.

  the

         vorticity(v(i,j),w(i,j))

      =  [-dv/dlambda + d(sint*w)/dtheta]/sint

      =  vort(i,j) - pertrb

  and

         divergence(v(i,j),w(i,j))

      =  [d(sint*v)/dtheta + dw/dlambda]/sint

      =  0.0

  where sint = sin(theta(i)).  required associated legendre polync
  are stored rather than recomputed as they are in subroutine ivrt



**************************************************************

  input parameters

  nlat   the number of colatitudes on the full sphere including th
         poles.
  nlon   the number of distinct longitude points.

  isym   this has the same value as the isym that was input to
         subroutine shaes to compute the arrays a and b.  isym
         determines whether (v,w) are computed on the full or half
         sphere as follows:

   = 0
         vort is not symmetric about the equator. in this case
         the vector field (v,w) is computed on the entire sphere.

  nt     in the program that calls ivrtes, nt is the number of vor
         and vector fields.

  idvw   the first dimension of the arrays v,w as it appears in
         the program that calls ivrtec.
```

```
        jdvw   the second dimension of the arrays v,w as it appears in
               the program that calls ivrtes.

        a,b    two or three dimensional arrays (see input parameter nt)
               that contain scalar spherical harmonic coefficients
               of the vorticity array vort as computed by subroutine sha
               a,b must be computed by shaes prior to calling ivrtes.

        mdab   the first dimension of the arrays a and b as it appears i
               the program that calls ivrtes (and shaes).

        ndab   the second dimension of the arrays a and b as it appears
               the program that calls ivrtes (and shaes).

        wvhses an array which must be initialized by subroutine vhsesi.

        lvhses the dimension of the array wvhses as it appears in the
               program that calls ivrtes.

        work   a work array that does not have to be saved.

        lwork  the dimension of the array work as it appears in the
               program that calls ivrtes.

      ************************************************************

        output parameters

        v,w    two or three dimensional arrays (see input parameter nt) t
               contain a divergence free vector field whose vorticity is
               vort - pertrb.

      pertrb   a nt dimensional array.

       ierror = 0  no errors
              = 1  error in the specification of nlat
              = 2  error in the specification of nlon
              = 3  error in the specification of isym
              = 4  error in the specification of nt
              = 5  error in the specification of idvw
              = 6  error in the specification of jdvw
              = 7  error in the specification of mdab
              = 8  error in the specification of ndab
              = 9  error in the specification of lvhses
              = 10 error in the specification of lwork
      ************************************************************
```

ivrtgc(...)

```
      ************************************************************

       given the scalar spherical harmonic coefficients a and b, precomp
```

```
by subroutine shagc for a scalar array vt, subroutine ivrtgc comp
a divergence free vector field (v,w) whose vorticity is vt - pert
w is the east longitude component and v is the colatitudinal comp
pertrb is a constant which must be subtracted from vt for (v,w) t
exist (see the description of pertrb below).  usually pertrb is z
or small relative to vort.  the divergence of (v,w), as computed
ivrtgc, is the zero scalar field.  v(i,j) and w(i,j) are the
colatitudinal and east longitude velocity components at gaussian
colatitude theta(i) (see nlat as input parameter) and longitude
lambda(j) = (j-1)*2*pi/nlon.  the

       vorticity(v(i,j),w(i,j))

    =  [-dv/dlambda + d(sint*w)/dtheta]/sint

    =  vort(i,j) - pertrb

and

       divergence(v(i,j),w(i,j))

    =  [d(sint*v)/dtheta + dw/dlambda]/sint

    =  0.0

where sint = sin(theta(i)).  required associated legendre polynom
are recomputed rather than stored as they are in subroutine ivrtg

*************************************************************

input parameters

nlat   the number of points in the gaussian colatitude grid on th
       full sphere.

nlon   the number of distinct londitude points.

isym   this has the same value as the isym that was input to
       subroutine shagc to compute the arrays a and b.  isym
       determines whether (v,w) are computed on the full or half
       sphere as follows:

 = 0
       vt is not symmetric about the equator. in this case
       the vector field (v,w) is computed on the entire sphere.

nt     in the program that calls ivrtgc, nt is the number of vort
       and vector fields.

idvw   the first dimension of the arrays v,w as it appears in
       the program that calls ivrtgc.
```

```
      jdvw   the second dimension of the arrays v,w as it appears in
             the program that calls ivrtgc.

      a,b    two or three dimensional arrays (see input parameter nt)
             that contain scalar spherical harmonic coefficients
             of the vorticity array vt as computed by subroutine shagc.
             a,b must be computed by shagc prior to calling ivrtgc.

      mdab   the first dimension of the arrays a and b as it appears in
             the program that calls ivrtgcs (and shagc).

      ndab   the second dimension of the arrays a and b as it appears i
             the program that calls ivrtgc (and shagc).

      wvhsg  an array which must be initialized by subroutine vhsgci.

      lvhsgc the dimension of the array wvhsgc as it appears in the
             program that calls ivrtgc.

      work   a work array that does not have to be saved.

      lwork  the dimension of the array work as it appears in the
             program that calls ivrtgc.

     ************************************************************

      output parameters

      v,w    two or three dimensional arrays (see input parameter nt) th
             contain a divergence free vector field whose vorticity is
             vt - pertrb.

     pertrb  a nt dimensional array.

     ierror = 0  no errors
            = 1  error in the specification of nlat
            = 2  error in the specification of nlon
            = 3  error in the specification of isym
            = 4  error in the specification of nt
            = 5  error in the specification of idvw
            = 6  error in the specification of jdvw
            = 7  error in the specification of mdab
            = 8  error in the specification of ndab
            = 9  error in the specification of lvhsgc
            = 10 error in the specification of lwork
     ************************************************************
```

ivrtgs(...)

```
     ************************************************************

      given the scalar spherical harmonic coefficients a and b, precomp
      by subroutine shags for a scalar array vt, subroutine ivrtgs comp
```

```
a divergence free vector field (v,w) whose vorticity is vt - pert
w is the east longitude component and v is the colatitudinal comp
pertrb is a constant which must be subtracted from vt for (v,w) t
exist (see the description of pertrb below).  usually pertrb is z
or small relative to vt.  the divergence of (v,w), as computed by
ivrtgs, is the zero scalar field.  v(i,j) and w(i,j) are the
colatitudinal and east longitude velocity components at gaussian
colatitude theta(i) (see nlat as input parameter) and longitude
lambda(j) = (j-1)*2*pi/nlon.  the

       vorticity(v(i,j),w(i,j))

    =  [-dv/dlambda + d(sint*w)/dtheta]/sint

    =  vort(i,j) - pertrb

and

       divergence(v(i,j),w(i,j))

    =  [d(sint*v)/dtheta + dw/dlambda]/sint

    =  0.0

where sint = sin(theta(i)).  required associated legendre polynom
are stored rather than recomputed as they are in subroutine ivrtg

*************************************************************

input parameters

nlat   the number of points in the gaussian colatitude grid on th
       full sphere.

nlon   the number of distinct londitude points.

isym   this has the same value as the isym that was input to
       subroutine shags to compute the arrays a and b.  isym
       determines whether (v,w) are computed on the full or half
       sphere as follows:
 = 0
       vt is not symmetric about the equator. in this case
       the vector field (v,w) is computed on the entire sphere.

nt     in the program that calls ivrtgs, nt is the number of vort
       and vector fields.

idvw   the first dimension of the arrays v,w as it appears in
       the program that calls ivrtgs.

jdvw   the second dimension of the arrays v,w as it appears in
       the program that calls ivrtgs.
```

```
      a,b    two or three dimensional arrays (see input parameter nt)
             that contain scalar spherical harmonic coefficients
             of the vorticity array vt as computed by subroutine shags.
             a,b must be computed by shags prior to calling ivrtgs.

      mdab   the first dimension of the arrays a and b as it appears in
             the program that calls ivrtgs (and shags).

      ndab   the second dimension of the arrays a and b as it appears i
             the program that calls ivrtgs (and shags).

      wvhsgs an array which must be initialized by subroutine vhsgsi.

      lvhsgs the dimension of the array wvhsgs as it appears in the
             program that calls ivrtgs.

      work   a work array that does not have to be saved.

      lwork  the dimension of the array work as it appears in the
             program that calls ivrtgs.

     ************************************************************

      output parameters


      v,w    two or three dimensional arrays (see input parameter nt) th
             contain a divergence free vector field whose vorticity is
             vt - pertrb.

      pertrb  a nt dimensional array

      ierror= 0  no errors
            = 1  error in the specification of nlat
            = 2  error in the specification of nlon
            = 3  error in the specification of isym
            = 4  error in the specification of nt
            = 5  error in the specification of idvw
            = 6  error in the specification of jdvw
            = 7  error in the specification of mdab
            = 8  error in the specification of ndab
            = 9  error in the specification of lvhsgs
            = 10 error in the specification of lwork
     ************************************************************
```

set_pyfort_option(...)

set_pyfort_option (value) sets default value of option keyword.

sfvpec(...)

```
     ************************************************************
```

```
given the vector spherical harmonic coefficients br,bi,cr,ci,
computed by subroutine vhaec for a vector field (v,w), sfvpec
computes a scalar stream function sf and scalar velocity potentia
vp for (v,w).  (v,w) is expressed in terms of sf and vp by the
helmholtz relations (in mathematical spherical coordinates):

     v = -1/sint*d(vp)/dlambda + d(st)/dtheta

     w =  1/sint*d(st)/dlambda + d(vp)/dtheta

where sint = sin(theta).  w is the east longitudinal and v
is the colatitudinal component of the vector field from which
br,bi,cr,ci were precomputed.  required associated legendre
polynomials are recomputed rather than stored as they are in
subroutine sfvpes. sf(i,j) and vp(i,j) are given at colatitude

       theta(i) = (i-1)*pi/(nlat-1)

and east longitude

       lambda(j) = (j-1)*2*pi/nlon

on the sphere.

*************************************************************

input parameters

nlat   the number of colatitudes on the full sphere including the
       poles.

nlon   the number of distinct longitude points.

isym   a parameter which determines whether the stream function a
       velocity potential are computed on the full or half sphere
       as follows:
 = 0
       the symmetries/antsymmetries described in isym=1,2 below
       do not exist in (v,w) about the equator.  in this case sf
       and vp are not necessarily symmetric or antisymmetric abou
       the equator.  sf and vp are computed on the entire sphere.

nt     nt is the number of scalar and vector fields.

idv    the first dimension of the arrays sf,vp as it appears in
       the program that calls sfvpec.

jdv    the second dimension of the arrays sf,vp as it appears in
       the program that calls sfvpec. jdv must be at least nlon.

br,bi, two or three dimensional arrays (see input parameter nt)
cr,ci  that contain vector spherical harmonic coefficients
```

```
             of the vector field (v,w) as computed by subroutine vhaec.

      mdb    the first dimension of the arrays br,bi,cr,ci as it
             appears in the program that calls sfvpec.

      ndb    the second dimension of the arrays br,bi,cr,ci as it
             appears in the program that calls sfvpec.

      wshsec an array which must be initialized by subroutine shseci.

      lshsec the dimension of the array wshsec as it appears in the
             program that calls sfvpec.

      work   a work array that does not have to be saved.

      lwork  the dimension of the array work as it appears in the

     ************************************************************

      output parameters

     sf,vp  two or three dimensional arrays (see input parameter nt)
            that contains the stream function and velocity potential
            of the vector field (v,w) whose coefficients br,bi,cr,ci
            where precomputed by subroutine vhaec.

     ierror = 0  no errors
            = 1  error in the specification of nlat
            = 2  error in the specification of nlon
            = 3  error in the specification of isym
            = 4  error in the specification of nt
            = 5  error in the specification of idv
            = 6  error in the specification of jdv
            = 7  error in the specification of mdb
            = 8  error in the specification of ndb
            = 9  error in the specification of lshsec
            = 10 error in the specification of lwork
     ************************************************************
```

sfvpes(...)

```
     ************************************************************

      given the vector spherical harmonic coefficients br,bi,cr,ci,
      computed by subroutine vhaes for a vector field (v,w), sfvpes
      computes a scalar stream function sf and scalar velocity potentia
      vp for (v,w).  (v,w) is expressed in terms of sf and vp by the
      helmholtz relations (in mathematical spherical coordinates):

           v = -1/sint*d(vp)/dlambda + d(st)/dtheta

           w =  1/sint*d(st)/dlambda + d(vp)/dtheta
```

```
where sint = sin(theta).  w is the east longitudinal and v
is the colatitudinal component of the vector field from which
br,bi,cr,ci were precomputed.  required associated legendre
polynomials are stored rather than recomputed as they are in
subroutine sfvpec. sf(i,j) and vp(i,j) are given at colatitude

       theta(i) = (i-1)*pi/(nlat-1)

and east longitude

       lambda(j) = (j-1)*2*pi/nlon

on the sphere.

*************************************************************

input parameters

nlat   the number of colatitudes on the full sphere including the

       poles.
nlon   the number of distinct longitude points.

isym   a parameter which determines whether the stream function a
       velocity potential are computed on the full or half sphere
       as follows:
 = 0
       the symmetries/antsymmetries described in isym=1,2 below
       do not exist in (v,w) about the equator.

nt     nt is the number of scalar and vector fields.

idv    the first dimension of the arrays sf,vp as it appears in
       the program that calls sfvpes.

jdv    the second dimension of the arrays sf,vp as it appears in
       the program that calls sfvpes.

br,bi, two or three dimensional arrays (see input parameter nt)
cr,ci  that contain vector spherical harmonic coefficients
       of the vector field (v,w) as computed by subroutine vhaec.

mdb    the first dimension of the arrays br,bi,cr,ci as it
       appears in the program that calls sfvpes.

ndb    the second dimension of the arrays br,bi,cr,ci as it
       appears in the program that calls sfvpes.

wshses an array which must be initialized by subroutine shsesi.

lshses the dimension of the array wshses as it appears in the
       program that calls sfrvpes.
```

```
 work   a work array that does not have to be saved.

 lwork  the dimension of the array work as it appears in the
        program that calls sfvpes.

***********************************************************

 output parameters

sf,vp  two or three dimensional arrays (see input parameter nt)
       that contains the stream function and velocity potential
       of the vector field (v,w) whose coefficients br,bi,cr,ci
       where computed by subroutine vhaec.

ierror = 0  no errors
       = 1  error in the specification of nlat
       = 2  error in the specification of nlon
       = 3  error in the specification of isym
       = 4  error in the specification of nt
       = 5  error in the specification of idv
       = 6  error in the specification of jdv
       = 7  error in the specification of mdb
       = 8  error in the specification of ndb
       = 9  error in the specification of lshses
       = 10 error in the specification of lwork
***********************************************************
```

sfvpgc(...)

```
***********************************************************


 given the vector spherical harmonic coefficients br,bi,cr,ci,
 computed by subroutine vhagc for a vector field (v,w), sfvpgc
 computes a scalar stream function sf and scalar velocity potentia
 vp for (v,w).  (v,w) is expressed in terms of sf and vp by the
 helmholtz relations (in mathematical spherical coordinates):

      v = -1/sint*d(vp)/dlambda + d(st)/dtheta

      w =  1/sint*d(st)/dlambda + d(vp)/dtheta

 where sint = sin(theta).  w is the east longitudinal and v
 is the colatitudinal component of the vector field from which
 br,bi,cr,ci were precomputed.  required associated legendre
 polynomials are recomputed rather than stored as they are in
 subroutine sfvpgs. sf(i,j) and vp(i,j) are given at the i(th)
 gaussian colatitude point theta(i) (see nlat description below)
 and east longitude lambda(j) = (j-1)*2*pi/nlon on the sphere.


***********************************************************
```

```
 input parameters

 nlat   the number of points in the gaussian colatitude grid on th
        full sphere.

 nlon   the number of distinct longitude points.

 isym   a parameter which determines whether the stream function a
        velocity potential are computed on the full or half sphere
        as follows:
  = 0
        the symmetries/antsymmetries described in isym=1,2 below
        do not exist in (v,w) about the equator.  in this case st
        and vp are not necessarily symmetric or antisymmetric abou
        the equator.  sf and vp are computed on the entire sphere.

 nt     nt is the number of scalar and vector fields.

 idv    the first dimension of the arrays sf,vp as it appears in
        the program that calls sfvpgc.

 jdv    the second dimension of the arrays sf,vp as it appears in
        the program that calls sfvpgc. jdv must be at least nlon.

 br,bi, two or three dimensional arrays (see input parameter nt)
 cr,ci  that contain vector spherical harmonic coefficients
        of the vector field (v,w) as computed by subroutine vhagc.

 mdb    the first dimension of the arrays br,bi,cr,ci as it
        appears in the program that calls sfvpgc.

 ndb    the second dimension of the arrays br,bi,cr,ci as it
        appears in the program that calls sfvpgc.

 wshsgc an array which must be initialized by subroutine shsgci.

 lshsgc the dimension of the array wshsgc as it appears in the
        program that calls sfvpgc.

 work   a work array that does not have to be saved.

 lwork  the dimension of the array work as it appears in the
        program that calls sfvpgc.

 ****************************************************************

 output parameters

sf,vp  two or three dimensional arrays (see input parameter nt)
       that contains the stream function and velocity potential
       of the vector field (v,w) whose coefficients br,bi,cr,ci
```

```
            where precomputed by subroutine vhagc.

      ierror = 0  no errors
             = 1  error in the specification of nlat
             = 2  error in the specification of nlon
             = 3  error in the specification of isym
             = 4  error in the specification of nt
             = 5  error in the specification of idv
             = 6  error in the specification of jdv
             = 7  error in the specification of mdb
             = 8  error in the specification of ndb
             = 9  error in the specification of lshsgc
             = 10 error in the specification of lwork
      ****************************************************************
```

sfvpgs(...)

```
      ****************************************************************

       given the vector spherical harmonic coefficients br,bi,cr,ci,
       computed by subroutine vhags for a vector field (v,w), sfvpgs
       computes a scalar stream function sf and scalar velocity potentia
       vp for (v,w).  (v,w) is expressed in terms of sf and vp by the
       helmholtz relations (in mathematical spherical coordinates):

            v = -1/sint*d(vp)/dlambda + d(st)/dtheta

            w =  1/sint*d(st)/dlambda + d(vp)/dtheta

       where sint = sin(theta).  w is the east longitudinal and v
       is the colatitudinal component of the vector field from which
       br,bi,cr,ci were precomputed.  required associated legendre
       polynomials are stored rather than recomputed as they are in
       subroutine sfvpgc. sf(i,j) and vp(i,j) are given at the i(th)
       gaussian colatitude point theta(i) (see nlat description below)
       and east longitude lambda(j) = (j-1)*2*pi/nlon on the sphere.

       ************************************************************

       input parameters

       nlat   the number of points in the gaussian colatitude grid on th
              full sphere.

       nlon   the number of distinct londitude points.

       isym   a parameter which determines whether the stream function a
              velocity potential are computed on the full or half sphere
              as follows:
        = 0
              the symmetries/antsymmetries described in isym=1,2 below
              do not exist in (v,w) about the equator.  in this case st
              and vp are not necessarily symmetric or antisymmetric abou
```

```
       the equator.

 nt    nt is the number of scalar and vector fields.

 idv   the first dimension of the arrays sf,vp as it appears in
       the program that calls sfvpgs.

 jdv   the second dimension of the arrays sf,vp as it appears in
       the program that calls sfvpgs.

 br,bi, two or three dimensional arrays (see input parameter nt)
 cr,ci  that contain vector spherical harmonic coefficients
       of the vector field (v,w) as computed by subroutine vhags.

 mdb   the first dimension of the arrays br,bi,cr,ci as it
       appears in the program that calls sfvpgs.

 ndb   the second dimension of the arrays br,bi,cr,ci as it
       appears in the program that calls sfvpgs.

 wshsgs an array which must be initialized by subroutine shsgsi.

 lshsgs the dimension of the array wshsgs as it appears in the
       program that calls sfvpgs.

 work  a work array that does not have to be saved.

 lwork  the dimension of the array work as it appears in the
       program that calls sfvpgs.

 ****************************************************************

 output parameters

sf,vp  two or three dimensional arrays (see input parameter nt)
       that contains the stream function and velocity potential
       of the vector field (v,w) whose coefficients br,bi,cr,ci
       where precomputed by subroutine vhags.

ierror = 0  no errors
       = 1  error in the specification of nlat
       = 2  error in the specification of nlon
       = 3  error in the specification of isym
       = 4  error in the specification of nt
       = 5  error in the specification of idv
       = 6  error in the specification of jdv
       = 7  error in the specification of mdb
       = 8  error in the specification of ndb
       = 9  error in the specification of lshsgs
       = 10 error in the specification of lwork
****************************************************************
```

```
      ************************************************************

      subroutine shaec performs the spherical harmonic analysis
      on the array g and stores the result in the arrays a and b.
      the analysis is performed on an equally spaced grid.  the
      associated legendre functions are recomputed rather than stored
      as they are in subroutine shaes.  the analysis is described
      below at output parameters a,b.

      ************************************************************

      input parameters

      nlat   the number of colatitudes on the full sphere including the
             poles.

      nlon   the number of distinct longitude points.

      isym   = 0  no symmetries exist about the equator. the analysis
                  is performed on the entire sphere.

      nt     the number of analyses.

      g      a two or three dimensional array (see input parameter
             nt) that contains the discrete function to be analyzed.

      idg    the first dimension of the array g as it appears in the
             program that calls shaec.

      jdg    the second dimension of the array g as it appears in the
             program that calls shaec.

      mdab   the first dimension of the arrays a and b as it appears
             in the program that calls shaec.
      ndab   the second dimension of the arrays a and b as it appears
             in the program that calls shaec.

      wshaec an array which must be initialized by subroutine shaeci.
             once initialized, wshaec can be used repeatedly by shaec
             as long as nlon and nlat remain unchanged.  wshaec must
             not be altered between calls of shaec.

      lshaec the dimension of the array wshaec as it appears in the
             program that calls shaec.

      work   a work array that does not have to be saved.

      lwork  the dimension of the array work as it appears in the
             program that calls shaec.

      ************************************************************
```

```
output parameters

a,b    both a,b are two or three dimensional arrays (see input
       parameter nt) that contain the spherical harmonic
       coefficients in the representation of g(i,j) given in the
       discription of subroutine shsec.

ierror = 0  no errors
       = 1  error in the specification of nlat
       = 2  error in the specification of nlon
       = 3  error in the specification of isym
       = 4  error in the specification of nt
       = 5  error in the specification of idg
       = 6  error in the specification of jdg
       = 7  error in the specification of mdab
       = 8  error in the specification of ndab
       = 9  error in the specification of lshaec
       = 10 error in the specification of lwork
*****************************************************************
```

shaeci(...)

```
*****************************************************************

subroutine shaeci initializes the array wshaec which can then
be used repeatedly by subroutine shaec.

*****************************************************************

input parameters

nlat   the number of colatitudes on the full sphere including the
       poles.

nlon   the number of distinct longitude points.

lshaec the dimension of the array wshaec as it appears in the
       program that calls shaeci.

dwork  a doubleprecision dwork array that does not have to be sav

ldwork the dimension of the array dwork as it appears in the
       program that calls shaeci.

*****************************************************************

output parameters

wshaec an array which is initialized for use by subroutine shaec.

ierror = 0  no errors
       = 1  error in the specification of nlat
```

```
            = 2  error in the specification of nlon
            = 3  error in the specification of lshaec
            = 4  error in the specification of ldwork
      ****************************************************************
```

shaes(...)

```
      ****************************************************************

       subroutine shaes performs the spherical harmonic analysis
       on the array g and stores the result in the arrays a and b.
       the analysis is performed on an equally spaced grid.  the
       associated legendre functions are stored rather than recomputed
       as they are in subroutine shaec.  the analysis is described
       below at output parameters a,b.


       ************************************************************

       input parameters

       nlat   the number of colatitudes on the full sphere including the
              poles.

       nlon   the number of distinct longitude points.

       isym   = 0  no symmetries exist about the equator. the analysis
                   is performed on the entire sphere.

       nt     the number of analyses.

       g      a two or three dimensional array (see input parameter
              nt) that contains the discrete function to be analyzed.

       idg    the first dimension of the array g as it appears in the
              program that calls shaes.

       jdg    the second dimension of the array g as it appears in the
              program that calls shaes.

       mdab   the first dimension of the arrays a and b as it appears
              in the program that calls shaes.

       ndab   the second dimension of the arrays a and b as it appears
              in the program that calls shaes.

       wshaes an array which must be initialized by subroutine shaesi.

       lshaes the dimension of the array wshaes as it appears in the
              program that calls shaes.

       work   a work array that does not have to be saved.
```

```
      lwork  the dimension of the array work as it appears in the
             program that calls shaes.


      ****************************************************************

      output parameters

      a,b    both a,b are two or three dimensional arrays (see input
             parameter nt) that contain the spherical harmonic
             coefficients in the representation of g(i,j) given in the
             discription of subroutine shses.

      ierror = 0  no errors
             = 1  error in the specification of nlat
             = 2  error in the specification of nlon
             = 3  error in the specification of isym
             = 4  error in the specification of nt
             = 5  error in the specification of idg
             = 6  error in the specification of jdg
             = 7  error in the specification of mdab
             = 8  error in the specification of ndab
             = 9  error in the specification of lshaes
             = 10 error in the specification of lwork
     ****************************************************************
```

shaesi(...)

```
     ****************************************************************


      subroutine shaesi initializes the array wshaes which can then
      be used repeatedly by subroutine shaes


      ****************************************************************

      input parameters

      nlat   the number of colatitudes on the full sphere including the
             poles.

      nlon   the number of distinct longitude points.

      lshaes the dimension of the array wshaes as it appears in the
             program that calls shaesi.

      work   a real   work array that does not have to be saved.

      lwork  the dimension of the array work as it appears in the
             program that calls shaesi.

      dwork  a doubleprecision work array that does not have to be save
```

```
      ldwork the dimension of the array dwork as it appears in the
             program that calls shaesi.

      ************************************************************

      output parameters

      wshaes an array which is initialized for use by subroutine shaes.

      ierror = 0  no errors
             = 1  error in the specification of nlat
             = 2  error in the specification of nlon
             = 3  error in the specification of lshaes
             = 4  error in the specification of lwork
             = 5  error in the specification of ldwork
     ************************************************************
```

shagc(...)

```
     ************************************************************

      subroutine shagc performs the spherical harmonic analysis
      on the array g and stores the result in the arrays a and b.
      the analysis is performed on a gaussian grid in colatitude
      and an equally spaced grid in longitude.  the associated
      legendre functions are recomputed rather than stored as they
      are in subroutine shags.  the analysis is described below
      at output parameters a,b.


      ************************************************************

      input parameters

      nlat   the number of points in the gaussian colatitude grid on th
             full sphere.

      nlon   the number of distinct longitude points.

      isym   = 0  no symmetries exist about the equator. the analysis
                  is performed on the entire sphere.

      nt     the number of analyses.

      g      a two or three dimensional array (see input parameter
             nt) that contains the discrete function to be analyzed.

      idg    the first dimension of the array g as it appears in the
             program that calls shagc.

      jdg    the second dimension of the array g as it appears in the
             program that calls shagc.
```

```
      mdab   the first dimension of the arrays a and b as it appears
             in the program that calls shagc.

      ndab   the second dimension of the arrays a and b as it appears
             in the program that calls shaec.

      wshagc an array which must be initialized by subroutine shagci.

      lshagc the dimension of the array wshagc as it appears in the
             program that calls shagc.

      work   a work array that does not have to be saved.

      lwork  the dimension of the array work as it appears in the
             program that calls shagc.


      ****************************************************************

      output parameters

      a,b    both a,b are two or three dimensional arrays (see input
             parameter nt) that contain the spherical harmonic
             coefficients in the representation of g(i,j) given in the
             discription of subroutine shagc.

      ierror = 0  no errors
             = 1  error in the specification of nlat
             = 2  error in the specification of nlon
             = 3  error in the specification of isym
             = 4  error in the specification of nt
             = 5  error in the specification of idg
             = 6  error in the specification of jdg
             = 7  error in the specification of mdab
             = 8  error in the specification of ndab
             = 9  error in the specification of lshagc
             = 10 error in the specification of lwork
     ****************************************************************
```

shagci(...)

```
     ****************************************************************

      subroutine shagci initializes the array wshagc which can then
      be used repeatedly by subroutines shagc. it precomputes
      and stores in wshagc quantities such as gaussian weights,
      legendre polynomial coefficients, and fft trigonometric tables.

      ****************************************************************

      input parameters
```

```
      nlat   the number of points in the gaussian colatitude grid on th
             full sphere.

      nlon   the number of distinct longitude points.

      wshagc an array which must be initialized by subroutine shagci.

      lshagc the dimension of the array wshagc as it appears in the
             program that calls shagc.

      dwork   a doubleprecision work array that does not have to be sav

      ldwork  the dimension of the array dwork as it appears in the
             program that calls shagci.

      ****************************************************************

      output parameter

      wshagc an array which must be initialized before calling shagc.

      ierror = 0  no errors
             = 1  error in the specification of nlat
             = 2  error in the specification of nlon
             = 3  error in the specification of lshagc
             = 4  error in the specification of ldwork
             = 5  failure in gaqd to compute gaussian points
                  (due to failure in eigenvalue routine)
     ****************************************************************
```

shags(...)

```
     ****************************************************************

      subroutine shags performs the spherical harmonic analysis
      on the array g and stores the result in the arrays a and b.
      the analysis is performed on a gaussian grid in colatitude
      and an equally spaced grid in longitude.  the associated
      legendre functions are stored rather than recomputed as they
      are in subroutine shagc.  the analysis is described below
      at output parameters a,b.


      ****************************************************************

      input parameters

      nlat   the number of points in the gaussian colatitude grid on th
             full sphere.

      nlon   the number of distinct longitude points.

      isym   = 0  no symmetries exist about the equator. the analysis
```

```
                is performed on the entire sphere.

     nt     the number of analyses.

     g      a two or three dimensional array (see input parameter
            nt) that contains the discrete function to be analyzed.

     idg    the first dimension of the array g as it appears in the

     jdg    the second dimension of the array g as it appears in the
            program that calls shags. jdg must be at least nlon.

     mdab   the first dimension of the arrays a and b as it appears
            in the program that calls shags.

     ndab   the second dimension of the arrays a and b as it appears
            in the program that calls shags.

     wshags an array which must be initialized by subroutine shagsi.

     lshags the dimension of the array wshags as it appears in the
            program that calls shags.

     work   a real work space which need not be saved

     lwork  the dimension of the array work as it appears in the
            program that calls shags.


     ****************************************************************

     output parameters

     a,b    both a,b are two or three dimensional arrays (see input
            parameter nt) that contain the spherical harmonic
            coefficients in the representation of g(i,j) given in the
            discription of subroutine shags.

     ierror = 0  no errors
            = 1  error in the specification of nlat
            = 2  error in the specification of nlon
            = 3  error in the specification of isym
            = 4  error in the specification of nt
            = 5  error in the specification of idg
            = 6  error in the specification of jdg
            = 7  error in the specification of mdab
            = 8  error in the specification of ndab
            = 9  error in the specification of lshags
            = 10 error in the specification of lwork
   ****************************************************************
```

shagsi(...)

```
      ************************************************************

       subroutine shagsi initializes the array wshags which can then
       be used repeatedly by subroutines shags. it precomputes
       and stores in wshags quantities such as gaussian weights,
       legendre polynomial coefficients, and fft trigonometric tables.


       ************************************************************

       input parameters

       nlat   the number of points in the gaussian colatitude grid on th
              full sphere.


       wshags an array which must be initialized by subroutine shagsi.

       lshags the dimension of the array wshags as it appears in the
              program that calls shags.

       work   a real work space which need not be saved

       lwork  the dimension of the array work as it appears in the
              program that calls shagsi.

       dwork   a doubleprecision work array that does not have to be sav

       ldwork  the length of dwork in the calling routine.


       ************************************************************

       output parameter

       wshags an array which must be initialized before calling shags.

       ierror = 0  no errors
              = 1  error in the specification of nlat
              = 2  error in the specification of nlon
              = 3  error in the specification of lshags
              = 4  error in the specification of lwork
              = 5  error in the specification of ldwork
              = 6  failure in gaqd to compute gaussian points
                   (due to failure in eigenvalue routine)
      ************************************************************
```

shigc(...)

```
      ************************************************************

       subroutine shigc initializes the array wshigc which can then
       be used repeatedly by subroutines shsgc or shagc. it precomputes
```

```
      and stores in wshigc quantities such as gaussian weights,
      legendre polynomial coefficients, and fft trigonometric tables.

      ************************************************************

      input parameters

      nlat   the number of points in the gaussian colatitude grid on th
             full sphere.

      nlon   the number of distinct londitude points.

      wshigc an array which must be initialized by subroutine shigc.

      lshigc the dimension of the array wshigc as it appears in the
             program that calls shsgc or shagc.

      dwork  a doubleprecision work array that does not have to be save

      ldwork the dimension of the array dwork as it appears in the
             program that calls shigc.

      ************************************************************

      output parameter

      wshigc an array which must be initialized before calling shsgc or

      ierror = 0  no errors
             = 1  error in the specification of nlat
             = 2  error in the specification of nlon
             = 3  error in the specification of lshigc
             = 4  error in the specification of ldwork
             = 5  failure in gaqd to compute gaussian points
                  (due to failure in eigenvalue routine)
    ************************************************************
```

shigs(...)

```
    ************************************************************

      subroutine shigs initializes the array wshigs which can then
      be used repeatedly by subroutines shags,shsgs. it precomputes
      and stores in wshigs quantities such as gaussian weights,
      legendre polynomial coefficients, and fft trigonometric tables.


      ************************************************************

      input parameters

      nlat   the number of points in the gaussian colatitude grid on th
             full sphere.
```

```
      nlon   the number of distinct longitude points. s

      wshigs an array which must be initialized by subroutine shigs .

      lshigs the dimension of the array wshigs as it appears in the
             program that calls shigs.

      work   a real work space which need not be saved

      lwork  the dimension of the array work as it appears in the
             program that calls shigs.

      dwork   a doubleprecision work array that does not have to be sav

      ldwork  the length of dwork in the calling routine.

      ************************************************************

      output parameter

      wshags an array which must be initialized before calling shags.

      ierror = 0  no errors
             = 1  error in the specification of nlat
             = 2  error in the specification of nlon
             = 3  error in the specification of lshags
             = 4  error in the specification of lwork
             = 5  error in the specification of ldwork
             = 6  failure in gaqd to compute gaussian points
                  (due to failure in eigenvalue routine)
     ****************************************************************
```

shsec(...)

```
     ****************************************************************

      subroutine shsec performs the spherical harmonic synthesis
      on the arrays a and b and stores the result in the array g.
      the synthesis is performed on an equally spaced grid.  the
      associated legendre functions are recomputed rather than stored
      as they are in subroutine shses.  the synthesis is described
      below at output parameter g.

      ************************************************************

      input parameters

      nlat   the number of colatitudes on the full sphere including the
             poles.

      nlon   the number of distinct longitude points.
```

```
isym   = 0  no symmetries exist about the equator. the synthesis
            is performed on the entire sphere.

nt     the number of syntheses.

idg    the first dimension of the array g as it appears in the
       program that calls shsec.

jdg    the second dimension of the array g as it appears in the
       program that calls shsec.  jdg must be at least nlon.

a,b    two or three dimensional arrays (see the input parameter
       nt) that contain the coefficients in the spherical harmoni
       expansion of g(i,j) given below at the definition of the
       output parameter g.

mdab   the first dimension of the arrays a and b as it appears
       in the program that calls shsec.

ndab   the second dimension of the arrays a and b as it appears
       in the program that calls shsec.

wshsec an array which must be initialized by subroutine shseci.

lshsec the dimension of the array wshsec as it appears in the
       program that calls shsec.

work   a work array that does not have to be saved.

lwork  the dimension of the array work as it appears in the
       program that calls shsec.

************************************************************

output parameters

g      a two or three dimensional array (see input parameter
       nt) that contains the spherical harmonic synthesis of
       the arrays a and b.
ierror = 0  no errors
       = 1  error in the specification of nlat
       = 2  error in the specification of nlon
       = 3  error in the specification of isym
       = 4  error in the specification of nt
       = 5  error in the specification of idg
       = 6  error in the specification of jdg
       = 7  error in the specification of mdab
       = 8  error in the specification of ndab
       = 9  error in the specification of lshsec
       = 10 error in the specification of lwork
************************************************************
```

shseci(...)

```
      ************************************************************

       subroutine shseci initializes the array wshsec which can then
       be used repeatedly by subroutine shsec.

       ************************************************************
       input parameters

       nlat   the number of colatitudes on the full sphere including the
              poles.

       nlon   the number of distinct longitude points.

       lshsec the dimension of the array wshsec as it appears in the
              program that calls shseci.

       dwork  a doubleprecision work array that does not have to be
              saved.

       ldwork the dimension of array dwork as it appears in the program
              that calls shseci.

       ************************************************************

       output parameters

       wshsec an array which is initialized for use by subroutine shsec.

       ierror = 0  no errors
              = 1  error in the specification of nlat
              = 2  error in the specification of nlon
              = 3  error in the specification of lshsec
              = 4  error in the specification of ldwork
      ************************************************************
```

shses(...)

```
      ************************************************************

       subroutine shses performs the spherical harmonic synthesis
       on the arrays a and b and stores the result in the array g.
       the synthesis is performed on an equally spaced grid.  the
       associated legendre functions are stored rather than recomputed
       as they are in subroutine shsec.  the synthesis is described
       below at output parameter g.

       ************************************************************

       input parameters

       nlat   the number of colatitudes on the full sphere including the
              poles.
```

```
nlon   the number of distinct longitude points.

isym   = 0  no symmetries exist about the equator. the synthesis
            is performed on the entire sphere.

nt     the number of syntheses.

idg    the first dimension of the array g as it appears in the
       program that calls shses.

jdg    the second dimension of the array g as it appears in the
       program that calls shses.

a,b    two or three dimensional arrays (see the input parameter
       nt) that contain the coefficients in the spherical harmoni
       expansion of g(i,j) given below at the definition of the
       output parameter g.

mdab   the first dimension of the arrays a and b as it appears
       in the program that calls shses.

ndab   the second dimension of the arrays a and b as it appears
       in the program that calls shses.

wshses an array which must be initialized by subroutine shsesi.

lshses the dimension of the array wshses as it appears in the

work   a work array that does not have to be saved.

lwork  the dimension of the array work as it appears in the
       program that calls shses.

**************************************************************

output parameters

g      a two or three dimensional array (see input parameter
       nt) that contains the spherical harmonic synthesis of
       the arrays a and b.

ierror = 0  no errors
       = 1  error in the specification of nlat
       = 2  error in the specification of nlon
       = 3  error in the specification of isym
       = 4  error in the specification of nt
       = 5  error in the specification of idg
       = 6  error in the specification of jdg
       = 7  error in the specification of mdab
       = 8  error in the specification of ndab
       = 9  error in the specification of lshses
```

```
             = 10 error in the specification of lwork
      ****************************************************************
```

shsesi(...)

```
      ****************************************************************

       subroutine shsesi initializes the array wshses which can then
       be used repeatedly by subroutine shses.

       ************************************************************

       input parameters

       nlat   the number of colatitudes on the full sphere including the
              poles.

       nlon   the number of distinct longitude points.

       lshses the dimension of the array wshses as it appears in the
              program that calls shsesi.

       work   a real   work array that does not have to be saved.

       lwork  the dimension of the array work as it appears in
              the program that calls shsesi.

       dwork  a doubleprecision work array that does not have to be save

       ldwork the dimension of the array dwork as it appears in the
              program that calls shsesi.  ldwork must be at least nlat+1

       ************************************************************

       output parameters

       wshses an array which is initialized for use by subroutine shses.

       ierror = 0  no errors
              = 1  error in the specification of nlat
              = 2  error in the specification of nlon
              = 3  error in the specification of lshses
              = 4  error in the specification of lwork
              = 5  error in the specification of ldwork
      ****************************************************************
```

shsgc(...)

```
      ****************************************************************

       subroutine shsgc performs the spherical harmonic synthesis
       on the arrays a and b and stores the result in the array g.
       the synthesis is performed on an equally spaced longitude grid
       and a gaussian colatitude grid.  the associated legendre function
```

are recomputed rather than stored as they are in subroutine
shsgs. the synthesis is described below at output parameter
g.

****

input parameters

nlat   the number of points in the gaussian colatitude grid on th
       full sphere.

nlon   the number of distinct longitude points.

mode   = 0 no symmetries exist about the equator. the synthesis
            is performed on the entire sphere.

nt   the number of syntheses.

idg   the first dimension of the array g as it appears in the
       program that calls shsgc.

jdg   the second dimension of the array g as it appears in the
       program that calls shsgc. jdg must be at least nlon.

mdab   the first dimension of the arrays a and b as it appears
       in the program that calls shsgc.

ndab   the second dimension of the arrays a and b as it appears
       in the program that calls shsgc.

a,b   two or three dimensional arrays (see the input parameter
       nt) that contain the coefficients in the spherical harmoni
       expansion of g(i,j) given below at the definition of the
       output parameter g.

wshsgc an array which must be initialized by subroutine shsgci.

lshsgc the dimension of the array wshsgc as it appears in the
       program that calls shsgc.

work   a work array that does not have to be saved.

lwork the dimension of the array work as it appears in the
       program that calls shsgc.

****

output parameters

g   a two or three dimensional array (see input parameter nt)

```
       that contains the discrete function which is synthesized.

 ierror = 0  no errors
       = 1  error in the specification of nlat
       = 2  error in the specification of nlon
       = 3  error in the specification of isym
       = 4  error in the specification of nt
       = 5  error in the specification of idg
       = 6  error in the specification of jdg
       = 7  error in the specification of mdab
       = 8  error in the specification of ndab
       = 9  error in the specification of lwshig
       = 10 error in the specification of lwork
     ****************************************************************
```

shsgci(...)

```
     ****************************************************************

      subroutine shsgci initializes the array wshsgc which can then
      be used repeatedly by subroutines shsgc. it precomputes
      and stores in wshsgc quantities such as gaussian weights,
      legendre polynomial coefficients, and fft trigonometric tables.


      ************************************************************

      input parameters

      nlat   the number of points in the gaussian colatitude grid on th
             full sphere.

      nlon   the number of distinct longitude points.

      wshsgc an array which must be initialized by subroutine shsgci.

      lshsgc the dimension of the array wshsgc as it appears in the
             program that calls shsgc.

      dwork  a doubleprecision work array that does not have to be save

      ldwork the dimension of the array dwork as it appears in the
             program that calls shsgci.


      ************************************************************

      output parameter

      wshsgc an array which must be initialized before calling shsgc.

      ierror = 0  no errors
             = 1  error in the specification of nlat
```

```
            = 2  error in the specification of nlon
            = 3  error in the specification of lshsgc
            = 4  error in the specification of ldwork
            = 5  failure in gaqd to compute gaussian points
                 (due to failure in eigenvalue routine)
     ************************************************************
```

shsgs(...)

```
     ************************************************************

      subroutine shsgs performs the spherical harmonic synthesis
      on the arrays a and b and stores the result in the array g.
      the synthesis is performed on an equally spaced longitude grid
      and a gaussian colatitude grid.  the associated legendre function
      are stored rather than recomputed as they are in subroutine
      shsgc.  the synthesis is described below at output parameter
      g.

      ************************************************************

      input parameters

      nlat   the number of points in the gaussian colatitude grid on th
             full sphere.

      nlon   the number of distinct longitude points.

      mode   = 0  no symmetries exist about the equator. the synthesis
                  is performed on the entire sphere.

      nt     the number of syntheses.

      idg    the first dimension of the array g as it appears in the
             program that calls shagc.

      jdg    the second dimension of the array g as it appears in the
             program that calls shagc.

      a,b    two or three dimensional arrays (see the input parameter
             nt) that contain the coefficients in the spherical harmoni
             expansion of g(i,j) given below at the definition of the
             output parameter g.

      mdab   the first dimension of the arrays a and b as it appears
             in the program that calls shsgs. mdab must be at least
             min0((nlon+2)/2,nlat) if nlon is even or at least
             min0((nlon+1)/2,nlat) if nlon is odd.

      ndab   the second dimension of the arrays a and b as it appears
             in the program that calls shsgs. ndab must be at least nla

      wshsgs an array which must be initialized by subroutine shsgsi.
```

```
      lshsgs the dimension of the array wshsgs as it appears in the
             program that calls shsgs.

      work   a work array that does not have to be saved.

      lwork  the dimension of the array work as it appears in the
             program that calls shsgs.

      ************************************************************

      output parameters

      g      a two or three dimensional array (see input parameter nt)
             that contains the discrete function which is synthesized.

      ierror = 0  no errors
             = 1  error in the specification of nlat
             = 2  error in the specification of nlon
             = 3  error in the specification of isym
             = 4  error in the specification of nt
             = 5  error in the specification of idg
             = 6  error in the specification of jdg
             = 7  error in the specification of mdab
             = 8  error in the specification of ndab
             = 9  error in the specification of lshsgs
             = 10 error in the specification of lwork
     ****************************************************************
```

shsgsi(...)

```
     ****************************************************************

      subroutine shsgsi initializes the array wshsgs which can then
      be used repeatedly by subroutines shsgs. it precomputes
      and stores in wshsgs quantities such as gaussian weights,
      legendre polynomial coefficients, and fft trigonometric tables.

      ************************************************************

      input parameters

      nlat   the number of points in the gaussian colatitude grid on th
             full sphere.

      nlon   the number of distinct longitude points.

      wshsgs an array which must be initialized by subroutine shsgsi.

      lshsgs the dimension of the array wshsgs as it appears in the
             program that calls shsgs.

      work   a real work space which need not be saved
```

```
      lwork  the dimension of the array work as it appears in the
             program that calls shsgsi. lwork must be at least
             4*nlat*(nlat+2)+2 in the routine calling shsgsi

      dwork   a doubleprecision work array that does not have to be sav

      ldwork  the length of dwork in the calling routine.

      ************************************************************

      output parameter

      wshsgs an array which must be initialized before calling shsgs.

      ierror = 0  no errors
             = 1  error in the specification of nlat
             = 2  error in the specification of nlon
             = 3  error in the specification of lshsgs
             = 4  error in the specification of lwork
             = 5  error in the specification of ldwork
             = 5  failure in gaqd to compute gaussian points
                  (due to failure in eigenvalue routine)
     ****************************************************************
```

slapec(...)

```
     ****************************************************************

      given the scalar spherical harmonic coefficients a and b, precomp
      by subroutine shaec for a scalar field sf, subroutine slapec comp
      the laplacian of sf in the scalar array slap.  slap(i,j) is the
      laplacian of sf at the colatitude

          theta(i) = (i-1)*pi/(nlat-1)

      and east longitude

          lambda(j) = (j-1)*2*pi/nlon

      on the sphere.  i.e.

          slap(i,j) =

                     2                   2
          [1/sint*d (sf(i,j)/dlambda + d(sint*d(sf(i,j))/dtheta)/dtheta

      where sint = sin(theta(i)).  the scalar laplacian in slap has the
      same symmetry or absence of symmetry about the equator as the sca
      field sf.  the input parameters isym,nt,mdab,ndab must have the
      same values used by shaec to compute a and b for sf. the associat
      legendre functions are recomputed rather than stored as they are
      in subroutine slapes.
```

```
*************************************************************

 input parameters

 nlat   the number of colatitudes on the full sphere including the
        poles.

 nlon   the number of distinct longitude points.

 isym   this parameter should have the same value input to subrout
        shaec to compute the coefficients a and b for the scalar f
        sf.  isym is set as follows:

        = 0  no symmetries exist in sf about the equator. scalar
             synthesis is used to compute slap on the entire spher

 nt     the number of analyses.

ids     the first dimension of the array slap as it appears in the
        program that calls slapec.

jds     the second dimension of the array slap as it appears in th
        program that calls slapec.


a,b     two or three dimensional arrays (see input parameter nt)
        that contain scalar spherical harmonic coefficients
        of the scalar field sf as computed by subroutine shaec.
        a,b must be computed by shaec prior to calling slapec.


mdab    the first dimension of the arrays a and b as it appears
        in the program that calls slapec.

ndab    the second dimension of the arrays a and b as it appears
        in the program that calls slapec.


wshsec  an array which must be initialized by subroutine shseci
        before calling slapec.

lshsec  the dimension of the array wshsec as it appears in the
        program that calls slapec.

 work   a work array that does not have to be saved.

 lwork  the dimension of the array work as it appears in the
        program that calls slapec.

 *************************************************************
```

```
 output parameters


slap    a two or three dimensional arrays (see input parameter nt)
        contain the scalar laplacian of the scalar field sf.

ierror    a parameter which flags errors in input parameters as fo
        = 0  no errors detected
        = 1  error in the specification of nlat
        = 2  error in the specification of nlon
        = 3  error in the specification of ityp
        = 4  error in the specification of nt
        = 5  error in the specification of ids
        = 6  error in the specification of jds
        = 7  error in the specification of mdbc
        = 8  error in the specification of ndbc
        = 9  error in the specification of lshsec
        = 10 error in the specification of lwork
*****************************************************************
```

***slapes*(...)**

```
*****************************************************************

 given the scalar spherical harmonic coefficients a and b, precomp
 by subroutine shaes for a scalar field sf, subroutine slapes comp
 the laplacian of sf in the scalar array slap.  slap(i,j) is the
 laplacian of sf at the colatitude

     theta(i) = (i-1)*pi/(nlat-1)

 and east longitude

     lambda(j) = (j-1)*2*pi/nlon

 on the sphere.  i.e.

     slap(i,j) =

     [1/sint*d (sf(i,j)/dlambda + d(sint*d(sf(i,j))/dtheta)/dtheta

 where sint = sin(theta(i)).  the scalar laplacian in slap has the
 same symmetry or absence of symmetry about the equator as the sca
 field sf.  the input parameters isym,nt,mdab,ndab must have the
 same values used by shaes to compute a and b for sf. the associat
 legendre functions are stored rather than recomputed as they are
 in subroutine slapec.

 *************************************************************

 input parameters

 nlat   the number of colatitudes on the full sphere including the
```

```
        poles.

 nlon   the number of distinct longitude points.

 isym   this parameter should have the same value input to subrout
        shaes to compute the coefficients a and b for the scalar f
        sf.  isym is set as follows:

        = 0  no symmetries exist in sf about the equator. scalar
             synthesis is used to compute slap on the entire spher

 nt     the number of analyses.

ids     the first dimension of the array slap as it appears in the
        program that calls slapes.

jds     the second dimension of the array slap as it appears in th
        program that calls slapes.

a,b     two or three dimensional arrays (see input parameter nt)
        that contain scalar spherical harmonic coefficients
        of the scalar field sf as computed by subroutine shaes.
        a,b must be computed by shaes prior to calling slapes.

mdab    the first dimension of the arrays a and b as it appears
        in the program that calls slapes.

ndab    the second dimension of the arrays a and b as it appears
        in the program that calls slapes.

wshses  an array which must be initialized by subroutine shsesi
        before calling slapes.

lshses  the dimension of the array wshses as it appears in the
        program that calls slapes.

 work   a work array that does not have to be saved.

 lwork  the dimension of the array work as it appears in the
        program that calls slapes.

 ****************************************************************

 output parameters

slap    a two or three dimensional arrays (see input parameter nt)
        contain the scalar laplacian of the scalar field sf.
        is the scalar laplacian at the colatitude

ierror    a parameter which flags errors in input parameters as fo
        = 0  no errors detected
        = 1  error in the specification of nlat
```

```
            = 2  error in the specification of nlon
            = 3  error in the specification of ityp
            = 4  error in the specification of nt
            = 5  error in the specification of ids
            = 6  error in the specification of jds
            = 7  error in the specification of mdbc
            = 8  error in the specification of ndbc
            = 9  error in the specification of lshses
            = 10 error in the specification of lwork
      ****************************************************************
```

slapgc(...)

```
      ****************************************************************

       given the scalar spherical harmonic coefficients a and b, precom
       by subroutine shagc for a scalar field sf, subroutine slapgc com
       the laplacian of sf in the scalar array slap.  slap(i,j) is the
       laplacian of sf at the gaussian colatitude theta(i) (see nlat as
       an input parameter) and east longitude lambda(j) = (j-1)*2*pi/nl
       on the sphere.  i.e.

           slap(i,j) =
                     2                   2
           [1/sint*d (sf(i,j)/dlambda + d(sint*d(sf(i,j))/dtheta)/dthet

       where sint = sin(theta(i)).  the scalar laplacian in slap has th
       same symmetry or absence of symmetry about the equator as the sc
       field sf.  the input parameters isym,nt,mdab,ndab must have the
       same values used by shagc to compute a and b for sf. the associa
       legendre functions are stored rather than recomputed as they are
       in subroutine slapgc.

      ****************************************************************

       input parameters

       nlat   the number of points in the gaussian colatitude grid on t
              full sphere.


       isym   this parameter should have the same value input to subrou
              shagc to compute the coefficients a and b for the scalar
              sf.  isym is set as follows:

              = 0  no symmetries exist in sf about the equator. scalar
                   synthesis is used to compute slap on the entire sphe

       nt     the number of analyses.

      ids     the first dimension of the array slap as it appears in th
              program that calls slapgc.
```

```
       jds     the second dimension of the array slap as it appears in t
               program that calls slapgc.

       a,b     two or three dimensional arrays (see input parameter nt)
               that contain scalar spherical harmonic coefficients
               of the scalar field sf as computed by subroutine shagc.
               a,b must be computed by shagc prior to calling slapgc.


       mdab    the first dimension of the arrays a and b as it appears
               in the program that calls slapgc.

       ndab    the second dimension of the arrays a and b as it appears
               in the program that calls slapgc.

               mdab,ndab should have the same values input to shagc to
               compute the coefficients a and b.

       wshsgc  an array which must be initialized by subroutine shsgci.

       lshsgc  the dimension of the array wshsgc as it appears in the
               program that calls slapgc.

        work   a work array that does not have to be saved.

        lwork  the dimension of the array work as it appears in the
               program that calls slapgc.

       ************************************************************

        output parameters

       slap    a two or three dimensional arrays (see input parameter nt
               contain the scalar laplacian of the scalar field sf.

      ierror    a parameter which flags errors in input parameters as fo
               = 0  no errors detected
               = 1  error in the specification of nlat
               = 2  error in the specification of nlon
               = 3  error in the specification of ityp
               = 4  error in the specification of nt
               = 5  error in the specification of ids
               = 6  error in the specification of jds
               = 7  error in the specification of mdbc
               = 8  error in the specification of ndbc
               = 9  error in the specification of lshsgc
               = 10 error in the specification of lwork
      ****************************************************************
```

slapgs(...)

```
      ****************************************************************
```

given the scalar spherical harmonic coefficients a and b, precom
by subroutine shags for a scalar field sf, subroutine slapgs com
the laplacian of sf in the scalar array slap. slap(i,j) is the
laplacian of sf at the gaussian colatitude theta(i) (see nlat as
an input parameter) and east longitude lambda(j) = (j-1)*2*pi/nl
on the sphere. i.e.

    slap(i,j) =
                2   2
    [1/sint*d (sf(i,j)/dlambda + d(sint*d(sf(i,j))/dtheta)/dthet

where sint = sin(theta(i)). the scalar laplacian in slap has th
same symmetry or absence of symmetry about the equator as the sc
field sf. the input parameters isym,nt,mdab,ndab must have the
same values used by shags to compute a and b for sf. the associa
legendre functions are stored rather than recomputed as they are
in subroutine slapgc.

****

input parameters

nlat   the number of points in the gaussian colatitude grid on t
       full sphere.

nlon   the number of distinct longitude points.

isym   this parameter should have the same value input to subrou
       shags to compute the coefficients a and b for the scalar
       sf. isym is set as follows:

       = 0 no symmetries exist in sf about the equator. scalar
            synthesis is used to compute slap on the entire sphe

nt   the number of analyses.

ids   the first dimension of the array slap as it appears in th
       program that calls slapgs.

jds   the second dimension of the array slap as it appears in t
       program that calls slapgs.

a,b   two or three dimensional arrays (see input parameter nt)
       that contain scalar spherical harmonic coefficients
       of the scalar field sf as computed by subroutine shags.
       a,b must be computed by shags prior to calling slapgs.

mdab   the first dimension of the arrays a and b as it appears
       in the program that calls slapgs.

ndab   the second dimension of the arrays a and b as it appears
       in the program that calls slapgs.

```
 wshsgs  an array which must be initialized by subroutine slapgsi
         (or equivalently by shsgsi).

 lshsgs  the dimension of the array wshsgs as it appears in the
         program that calls slapgs.

  work   a work array that does not have to be saved.

  lwork  the dimension of the array work as it appears in the
         program that calls slapgs.

 ************************************************************

  output parameters

 slap    a two or three dimensional arrays (see input parameter nt
         contain the scalar laplacian of the scalar field sf.

 ierror    a parameter which flags errors in input parameters as f
         = 0  no errors detected
         = 1  error in the specification of nlat
         = 2  error in the specification of nlon
         = 3  error in the specification of ityp
         = 4  error in the specification of nt
         = 5  error in the specification of ids
         = 6  error in the specification of jds
         = 7  error in the specification of mdbc
         = 8  error in the specification of ndbc
         = 9  error in the specification of lshsgs
         = 10 error in the specification of lwork
*****************************************************************
```

sshifte(...)

```
*****************************************************************

  subroutine sshifte does a highly accurate 1/2 grid increment shi
  in both longitude and latitude of equally spaced data on the sph
  data is transferred between the nlon by nlat offset grid in goff
  (which excludes poles) and the nlon by nlat+1 regular grid in gr
  (which includes poles).  the transfer can go from goff to greg o
  greg to goff (see ioff).  the grids which underly goff and greg
  described below.  the north and south poles are at latitude 0.5*
  -0.5*pi radians respectively where pi = 4.*atan(1.).


 ************************************************************

  input parameters

  nlon   the number of longitude points on both the offset and reg
         uniform grid in longitude.
```

```
        nlat    the number of latitude points on the offset uniform grid
                is the number of latitude points on the regular uniform

        greg    a nlon by nlat+1 array that contains input data on the r
                described above.

        goff    a nlon by nlat array that contains input data on the off
                described above.

        wsav    a real saved work space array that must be initialized b
                subroutine sshift2regi(nlon,nlat,wsav,ier) before callin

        lsav    the length of the saved work space wsav in the routine c
                and sshiftei.

        wrk     a real unsaved work space

        lwrk    the length of the unsaved work space in the routine call

      ************************************************************

       output parameters

        ier = 0 if no errors are detected
            = 1 if ioff is not equal to 0 or 1
            = 1 if nlon < 4
            = 2 if nlat < 3
            = 3 if lsave < 2*(nlon+2*nlat+16)
            = 4 if lwork < 2*nlon*(nlat+1) for nlon even or
                lwork < nlon*(5*nlat+1) for nlon odd
      ************************************************************
```

sshifti(...)

```
      ************************************************************

        subroutine sshifti initializes the saved work space wsav
        for ioff and nlon and nlat (see documentation for sshifte).
        sshifti must be called before sshifte whenever ioff or nlon
        or nlat change.

        ier = 0 if no errors with input arguments
            = 1 if ioff is not 0 or 1
            = 2 if nlon < 4
            = 3 if nlat < 3
            = 4 if lsav < 2*(2*nlat+nlon+16)
      ************************************************************
```

trssph(...)

```
      ************************************************************

        subroutine trssph transfers data given in array da on a grid on
```

full sphere to data in array db on a grid on the full sphere. t
grids on which da is given and db is generated can be specified
independently of each other (see description below and the argum
igrida,igridb). for transferring vector data on the sphere, use
subroutine trvsph.

notice that scalar and vector quantities are fundamentally diffe
on the sphere. for example, vectors are discontinuous and multi
valued at the poles. scalars are continuous and single valued a
poles. erroneous results would be produced if one attempted to t
vector fields between grids with subroutine trssph applied to ea
component of the vector.

****

input arguments

intl — an initialization argument which should be zero on
trssph.

igrida — an integer vector dimensioned two which identifies
on the full sphere for the given data array da as f

igrida(1)

= -1 — if the latitude (or colatitude) grid for da is an e
partition of [-pi/2,pi/2] (or [0,pi]) including t
runs north to south

= +1 — if the latitude (or colatitude) grid for da is an e
partition of [-pi/2,pi/2] (or [0,pi]) including t
runs south to north

= -2 — if the latitude (or colatitude) grid for da is a ga
of (-pi/2,pi/2) (or (0,pi)) excluding the poles w
to south

= +2 — if the latitude (or colatitude) grid for da is a ga
of (-pi/2,pi/2) (or (0,pi)) excluding the poles w
north

igrida(2)

= 0 — if the underlying grid for da is a nlona by nlata

= 1 — if the underlying grid for da is a nlata by nlona

nlona — the number of longitude points on the uniform grid
[0,2pi) for the given data array da.

nlata — the number of points in the latitude (or colatitude
for the given data array da.

```
  da           a two dimensional array that contains the data to b

  igridb       an integer vector dimensioned two which identifies
               on the full sphere for the transformed data array d

  igridb(1)

      = -1     if the latitude (or colatitude) grid for db is an e
               partition of [-pi/2,pi/2] ( or [0,pi] ) including t
               north to south

      = +1     if the latitude (or colatitude) grid for db is an e
               partition of [-pi/2,pi/2] ( or [0,pi] ) including t
               south to north

      = -2     if the latitude (or colatitude) grid for db is a ga
               of (-pi/2,pi/2) ( or (0,pi) ) excluding the poles w
               south

      = +2     if the latitude (or colatitude) grid for db is a ga
               of (-pi/2,pi/2) ( or (0,pi) ) excluding the poles w
               north
  igridb(2)

       = 0     if the underlying grid for db is a nlonb by nlatb

       = 1     if the underlying grid for db is a nlatb by nlonb


  nlonb        the number of longitude points on the uniform grid
               [0,2pi) for the transformed data array db.

  nlatb        the number of points in the latitude (or colatitude
               for the transformed data array db.

  wsave        a saved work space array that can be utilized repea
               as long as the arguments nlata,nlona,nlatb,nlonb re

  lsave        the dimension of the work space wsave as it appears
               that calls trssph.

  work         a real work array that does not have to be preserve

  lwork        the dimension of the array work as it appears in th
               calling trssph.

  dwork        a doubleprecision work array that does not have to

  ldwork       the length of dwork in the routine calling trssph.

 ************************************************************
```

```
  output arguments

  db           a two dimensional array that contains the transform

  lsvmin       the minimum length of the saved work space in wsave

  lwkmin       the minimum length of the unsaved work space in wor

  ier = 0  if no errors are detected
      = 1  if intl is not 0 or 1
      = 2  if igrida(1) is not -1 or +1 or -2 or +2
      = 3  if igrida(2) is not 0 or 1
      = 4  if nlona is less than 4
      = 5  if nlata is less than 3
      = 6  if igridb(1) is not -1 or +1 or -2 or +2
      = 7  if igridb(2) is not 0 or 1
      = 8  if nlonb is less than 4
      = 9  if nlatb is less than 3
      =10  if there is insufficient saved work space (lsave < lsvm
      =11  if there is insufficient unsaved work space (lwork < lw
      =12  indicates failure in an eigenvalue routine which comput
           gaussian weights and points
      =13  if ldwork is too small (insufficient unsaved doubleprec
           work space)
****************************************************************
```

trvsph(...)

```
****************************************************************

  subroutine trvsph transfers vector data given in (ua,va) on a gr
  the full sphere to vector data in (ub,vb) on a grid on the full
  the grids on which (ua,va) is given and (ub,vb) is generated can
  specified independently of each other (see the input arguments i
  igridb,iveca,ivecb).  ua and ub are the east longitudinal compon
  the given and transformed vector fields.  va is either the latit
  or colatitudinal component of the given vector field (see iveca)
  vb is either the latitudinal or colatitudinal component of the
  transformed vector field (see ivecb).  for transferring scalar d
  on the sphere, use subroutine trssph.

  notice that scalar and vector quantities are fundamentally diffe
  on the sphere.  for example, vectors are discontinuous and multi
  valued at the poles.  scalars are continuous and single valued a
  poles. erroneous results would be produced if one attempted to t
  vector fields between grids with subroutine trssph applied to ea
  component of the vector.

****************************************************************

  input arguments
```

intl an initialization argument which should be zero on
trvsph.

igrida an integer vector dimensioned two which identifies
on the full sphere for the given vector data (ua,va

igrida(1)

= -1 if the latitude (or colatitude) grid for ua,va is a
partition of [-pi/2,pi/2] (or [0,pi]) including t
runs north to south with increasing subscript value

= +1 if the latitude (or colatitude) grid for ua,va is a
partition of [-pi/2,pi/2] (or [0,pi]) including t
runs south to north with increasing subscript value

= -2 if the latitude (or colatitude) grid for ua,va is a
of (-pi/2,pi/2) (or (0,pi)) excluding the poles w
to south with increasing subscript value

= +2 if the latitude (or colatitude) grid for ua,va is a
of (-pi/2,pi/2) (or (0,pi)) excluding the poles w
north with increasing subscript value

igrida(2)

= 0 if the underlying grid for ua,va is a nlona by nlat

= 1 if the underlying grid for ua,va is a nlata by nlon

nlona the number of longitude points on the uniform grid
[0,2pi) for the given vector (ua,va).

nlata the number of points in the latitude (or colatitude
given vector (ua,va).

iveca if iveca=0 is input then va is the latitudinal comp
given vector field. if iveca=1 then va is the colat
compoenent of the given vector field.

ua ua is the east longitudinal component of the given

va va is either the latitudinal or colatitudinal compo
given vector field (see iveca).

igridb an integer vector dimensioned two which identifies
on the full sphere for the transformed vector (ub,v

igridb(1)

= -1 if the latitude (or colatitude) grid for ub,vb is a
partition of [-pi/2,pi/2] (or [0,pi]) including t
north to south

```
          = +1     if the latitude (or colatitude) grid for ub,vb is a
                   partition of [-pi/2,pi/2] ( or [0,pi] ) including t
                   south to north

          = -2     if the latitude (or colatitude) grid for ub,vb is a
                   of (-pi/2,pi/2) ( or (0,pi) ) excluding the poles w
                   south

          = +2
                   if the latitude (or colatitude) grid for ub,vb is a
                   of (-pi/2,pi/2) ( or (0,pi) ) excluding the poles w
                   north
    igridb(2)

          = 0      if the underlying grid for ub,vb is a nlonb by nlat

          = 1      if the underlying grid for ub,vb is a nlatb by nlon

    nlonb         the number of longitude points on the uniform grid
                  [0,2pi) for the transformed vector (ub,vb).

    nlatb         the number of points in the latitude (or colatitude
                  transformed vector (ub,vb).

    ivecb         if ivecb=0 is input then vb is the latitudinal comp
                  given vector field. if ivecb=1 then vb is the colat
                   component of the given vector field.

    wsave         a saved work space array that can be utilized repea
                  as long as the arguments nlata,nlona,nlatb,nlonb re

    lsave         the dimension of the work space wsave as it appears
                  that calls trvsph.

    work          a work array that does not have to be preserved

    lwork         the dimension of the array work as it appears in th
                  calls trvsph.

    dwork         a doubleprecision work array that does not have to

    ldwork        the length of dwork in the routine calling trvsph

  ************************************************************

    output arguments

    ub            a two dimensional array that contains the east long
                  of the transformed vector data.

    vb            a two dimensional array that contains the latitudin
```

```
                  component of the transformed vector data (see ivecb

   lsvmin         the minimum length of the saved work space in wsave
                  lsvmin is computed even if lsave < lsvmin (ier = 10

   lwkmin         the minimum length of the unsaved work space in wor
                  lwkmin is computed even if lwork < lwkmin (ier = 11

  ier = 0  if no errors are detected
       = 1  if intl is not 0 or 1
       = 2  if igrida(1) is not -1 or +1 or -2 or +2
       = 3  if igrida(2) is not 0 or 1
       = 4  if nlona is less than 4
       = 5  if nlata is less than 3
       = 6  if iveca is not 0 or 1
       = 7  if igridb(1) is not -1 or +1 or -2 or +2
       = 8  if igridb(2) is not 0 or 1
       = 9  if nlonb is less than 4
       =10  if nlatb is less than 3
       =11  if ivecb is not 0 or 1
       =12  if there is insufficient saved work space (lsave < lsvm
       =13  if there is insufficient unsaved work space (lwork < lw
       =14  indicates failure in an eigenvalue routine which comput
            gaussian weights and points
       =15  if ldwork is too small (insufficient doubleprecision
            unsaved work space)
*****************************************************************
```

vhaec(...)

```
*****************************************************************

subroutine vhaec performs the vector spherical harmonic analysis
on the vector field (v,w) and stores the result in the arrays
br, bi, cr, and ci. v(i,j) and w(i,j) are the colatitudinal
(measured from the north pole) and east longitudinal components
respectively, located at colatitude theta(i) = (i-1)*pi/(nlat-1)
and longitude phi(j) = (j-1)*2*pi/nlon. the spectral
representation of (v,w) is given at output parameters v,w in
subroutine vhsec.

*************************************************************

input parameters

nlat   the number of colatitudes on the full sphere including the
       poles.

nlon   the number of distinct longitude points.

ityp   = 0  no symmetries exist about the equator.

nt     the number of analyses.
```

```
      v,w    two or three dimensional arrays (see input parameter nt)
             that contain the vector function to be analyzed.

      idvw   the first dimension of the arrays v,w as it appears in
             the program that calls vhaec.

      jdvw   the second dimension of the arrays v,w as it appears in
             the program that calls vhaec.

      mdab   the first dimension of the arrays br,bi,cr, and ci as it
             appears in the program that calls vhaec.

      ndab   the second dimension of the arrays br,bi,cr, and ci as it
             appears in the program that calls vhaec.

      wvhaec an array which must be initialized by subroutine vhaeci.

      lvhaec the dimension of the array wvhaec as it appears in the
             program that calls vhaec.

      work   a work array that does not have to be saved.

      lwork  the dimension of the array work as it appears in the
             program that calls vhaec.

      ****************************************************************

      output parameters

      br,bi  two or three dimensional arrays (see input parameter nt)
      cr,ci  that contain the vector spherical harmonic coefficients
             in the spectral representation of v(i,j) and w(i,j) given
             in the discription of subroutine vhsec.

      ierror = 0  no errors
             = 1  error in the specification of nlat
             = 2  error in the specification of nlon
             = 3  error in the specification of ityp
             = 4  error in the specification of nt
             = 5  error in the specification of idvw
             = 6  error in the specification of jdvw
             = 7  error in the specification of mdab
             = 8  error in the specification of ndab
             = 9  error in the specification of lvhaec
             = 10 error in the specification of lwork
      ****************************************************************
```

vhaeci(...)

```
      ****************************************************************

       subroutine vhaeci initializes the array wvhaec which can then be
```

```
      used repeatedly by subroutine vhaec until nlat or nlon is changed


     ************************************************************

      input parameters

      nlat   the number of colatitudes on the full sphere including the
             poles.
      nlon   the number of distinct londitude points.

      dwork  a doubleprecision work array that does not have to be save

      ldwork the dimension of the array dwork as it appears in the
             program that calls vhaec.

     ************************************************************

      output parameters

      wvhaec an array which is initialized for use by subroutine vhaec.
             once initialized, wvhaec can be used repeatedly by vhaec
             as long as nlat or nlon remain unchanged.  wvhaec must not
             be altered between calls of vhaec.

      ierror = 0  no errors
             = 1  error in the specification of nlat
             = 2  error in the specification of nlon
             = 3  error in the specification of lvhaec
             = 4  error in the specification of ldwork
     ************************************************************
```

vhaes(...)

```
     ************************************************************

     subroutine vhaes performs the vector spherical harmonic analysis
     on the vector field (v,w) and stores the result in the arrays
     br, bi, cr, and ci. v(i,j) and w(i,j) are the colatitudinal
     (measured from the north pole) and east longitudinal components
     respectively, located at colatitude theta(i) = (i-1)*pi/(nlat-1)
     and longitude phi(j) = (j-1)*2*pi/nlon. the spectral
     representation of (v,w) is given at output parameters v,w in
     subroutine vhses.

     ************************************************************

     input parameters

     nlat   the number of colatitudes on the full sphere including the
            poles.

     nlon   the number of distinct londitude points.
```

```
ityp   = 0  no symmetries exist about the equator. the analysis
            is performed on the entire sphere.

nt     the number of analyses.

v,w    two or three dimensional arrays (see input parameter nt)
       that contain the vector function to be analyzed.
       v is the colatitudnal component and w is the east
       longitudinal component.

idvw   the first dimension of the arrays v,w as it appears in
       the program that calls vhaes.

jdvw   the second dimension of the arrays v,w as it appears in
       the program that calls vhaes.

mdab   the first dimension of the arrays br,bi,cr, and ci as it
       appears in the program that calls vhaes.

ndab   the second dimension of the arrays br,bi,cr, and ci as it
       appears in the program that calls vhaes.
lvhaes an array which must be initialized by subroutine vhaesi.

lvhaes the dimension of the array wvhaes as it appears in the
       program that calls vhaes.

work   a work array that does not have to be saved.

lwork  the dimension of the array work as it appears in the
       program that calls vhaes.

************************************************************

output parameters

br,bi  two or three dimensional arrays (see input parameter nt)
cr,ci  that contain the vector spherical harmonic coefficients
       in the spectral representation of v(i,j) and w(i,j) given
       in the discription of subroutine vhses.

ierror = 0  no errors
       = 1  error in the specification of nlat
       = 2  error in the specification of nlon
       = 3  error in the specification of ityp
       = 4  error in the specification of nt
       = 5  error in the specification of idvw
       = 6  error in the specification of jdvw
       = 7  error in the specification of mdab
       = 8  error in the specification of ndab
       = 9  error in the specification of lvhaes
       = 10 error in the specification of lwork
```

```
      ************************************************************************

vhaesi(...)
      ************************************************************************

       subroutine vhaesi initializes the array wvhaes which can then be
       used repeatedly by subroutine vhaes until nlat or nlon is changed

      ************************************************************

       input parameters

       nlat   the number of colatitudes on the full sphere including the
              poles.

       nlon   the number of distinct londitude points.

       lvhaes the dimension of the array wvhaes as it appears in the
              program that calls vhaes.

       work   a work array that does not have to be saved.

       lwork  the dimension of the array work as it appears in the
              program that calls vhaes.

       dwork  an unsaved doubleprecision work space

       ldwork the length of the array dwork as it appears in the
              program that calls vhaesi.

      ************************************************************

       output parameters

       wvhaes an array which is initialized for use by subroutine vhaes.

       ierror = 0  no errors
              = 1  error in the specification of nlat
              = 2  error in the specification of nlon
              = 3  error in the specification of lvhaes
              = 4  error in the specification of lwork
              = 5  error in the specification of ldwork
      ************************************************************************

vhagc(...)
      ************************************************************************

       subroutine vhagc performs the vector spherical harmonic analysis
       on the vector field (v,w) and stores the result in the arrays
       br,bi,cr, and ci. v(i,j) and w(i,j) are the colatitudinal
       (measured from the north pole) and east longitudinal components
       respectively, located at the gaussian colatitude point theta(i)
```

```
 and longitude phi(j) = (j-1)*2*pi/nlon. the spectral
 representation of (v,w) is given at output parameters v,w in
 subroutine vhsec.

*************************************************************

 input parameters

 nlat   the number of points in the gaussian colatitude grid on th
        full sphere.

 nlon   the number of distinct longitude points.

 ityp   = 0  no symmetries exist about the equator. the analysis
             is performed on the entire sphere.

 nt     the number of analyses.

 v,w    two or three dimensional arrays (see input parameter nt)
        that contain the vector function to be analyzed.
        the program that calls vhagc.

 jdvw   the second dimension of the arrays v,w as it appears in
        the program that calls vhagc.

 mdab   the first dimension of the arrays br,bi,cr, and ci as it
        appears in the program that calls vhagc.

 ndab   the second dimension of the arrays br,bi,cr, and ci as it
        appears in the program that calls vhagc.

 wvhagc an array which must be initialized by subroutine vhagci.

 lvhagc the dimension of the array wvhagc as it appears in the
        program that calls vhagc.

 lwork  the dimension of the array work as it appears in the
        program that calls vhagc.

*************************************************************

 output parameters

 br,bi  two or three dimensional arrays (see input parameter nt)
 cr,ci  that contain the vector spherical harmonic coefficients
        in the spectral representation of v(i,j) and w(i,j) given
        in the discription of subroutine vhsec.

 ierror = 0  no errors
        = 1  error in the specification of nlat
        = 2  error in the specification of nlon
        = 3  error in the specification of ityp
```

```
            = 4  error in the specification of nt
            = 5  error in the specification of idvw
            = 6  error in the specification of jdvw
            = 7  error in the specification of mdab
            = 8  error in the specification of ndab
            = 9  error in the specification of lvhagc
            = 10 error in the specification of lwork
     ****************************************************************
```

vhagci(...)

```
     ****************************************************************

      subroutine vhagci initializes the array wvhagc which can then be
      used repeatedly by subroutine vhagc until nlat or nlon is changed

     ************************************************************

      input parameters

      nlat   the number of points in the gaussian colatitude grid on th
             full sphere.

      nlon   the number of distinct londitude points.

      lvhagc the dimension of the array wvhagc as it appears in the
             program that calls vhagci.

      dwork  a doubleprecision work array that does not need to be save

      ldwork the dimension of the array dwork as it appears in the
             program that calls vhagci.

     ************************************************************

      output parameters

      wvhagc an array which is initialized for use by subroutine vhagc.

      ierror = 0  no errors
             = 1  error in the specification of nlat
             = 2  error in the specification of nlon
             = 3  error in the specification of lvhagc
             = 4  error in the specification of lwork
     ****************************************************************
```

vhags(...)

```
     ****************************************************************

      subroutine vhags performs the vector spherical harmonic analysis
      on the vector field (v,w) and stores the result in the arrays
      br, bi, cr, and ci. v(i,j) and w(i,j) are the colatitudinal
      (measured from the north pole) and east longitudinal components
```

```
 respectively, located at the gaussian colatitude point theta(i)
 and longitude phi(j) = (j-1)*2*pi/nlon. the spectral
 representation of (v,w) is given at output parameters v,w in
 subroutine vhses.

*************************************************************

 input parameters

 nlat   the number of points in the gaussian colatitude grid on th
        full sphere.

 nlon   the number of distinct londitude points.

 ityp   = 0  no symmetries exist about the equator. the analysis
             is performed on the entire sphere.

 nt     the number of analyses.

 v,w    two or three dimensional arrays (see input parameter nt)
        that contain the vector function to be analyzed.

 idvw   the first dimension of the arrays v,w as it appears in
        the program that calls vhags.

 jdvw   the second dimension of the arrays v,w as it appears in
        the program that calls vhags.

 mdab   the first dimension of the arrays br,bi,cr, and ci as it
        appears in the program that calls vhags.

 ndab   the second dimension of the arrays br,bi,cr, and ci as it
        appears in the program that calls vhags.

 wvhags an array which must be initialized by subroutine vhgsi.

 lvhags the dimension of the array wvhags as it appears in the
        program that calls vhags.

 work   a work array that does not have to be saved.

 lwork  the dimension of the array work as it appears in the
        program that calls vhags.

*************************************************************

 output parameters

 br,bi  two or three dimensional arrays (see input parameter nt)
 cr,ci  that contain the vector spherical harmonic coefficients
        in the spectral representation of v(i,j) and w(i,j) given
        in the discription of subroutine vhses.
```

```
 ierror = 0  no errors
        = 1  error in the specification of nlat
        = 2  error in the specification of nlon
        = 3  error in the specification of ityp
        = 4  error in the specification of nt
        = 5  error in the specification of idvw
        = 6  error in the specification of jdvw
        = 7  error in the specification of mdab
        = 8  error in the specification of ndab
        = 9  error in the specification of lvhags
        = 10 error in the specification of lwork
      ************************************************************
```

vhagsi(...)

```
      ************************************************************

       subroutine vhagsi initializes the array wvhags which can then be
       used repeatedly by subroutine vhags until nlat or nlon is changed

      ************************************************************

       input parameters

       nlat   the number of points in the gaussian colatitude grid on th
              full sphere.

       nlon   the number of distinct longitude points.

       lvhags the dimension of the array wvhags as it appears in the
              program that calls vhagsi.

       dwork  a doubleprecision work space that does not need to be save

       ldwork the dimension of the array dwork as it appears in the
              program that calls vhagsi.

      ************************************************************

       output parameters

       wvhags an array which is initialized for use by subroutine vhags.

       ierror = 0  no errors
              = 1  error in the specification of nlat
              = 2  error in the specification of nlon
              = 3  error in the specification of lvhags
              = 4  error in the specification of ldwork
      ************************************************************
```

vhsec(...)

```
*************************************************************

 subroutine vhsec performs the vector spherical harmonic synthesis
 of the arrays br, bi, cr, and ci and stores the result in the
 arrays v and w. v(i,j) and w(i,j) are the colatitudinal
 (measured from the north pole) and east longitudinal components
 respectively, located at colatitude theta(i) = (i-1)*pi/(nlat-1)
 and longitude phi(j) = (j-1)*2*pi/nlon. the spectral
 representation of (v,w) is given below at output parameters v,w.

*************************************************************

 input parameters

 nlat   the number of colatitudes on the full sphere including the
        poles.

 nlon   the number of distinct longitude points.

 ityp   = 0  no symmetries exist about the equator. the synthesis
             is performed on the entire sphere.

 nt     the number of syntheses.

 idvw   the first dimension of the arrays v,w as it appears in
        the program that calls vhsec.

 jdvw   the second dimension of the arrays v,w as it appears in
        the program that calls vhsec.

 br,bi  two or three dimensional arrays (see input parameter nt)
 cr,ci  that contain the vector spherical harmonic coefficients
        in the spectral representation of v(i,j) and w(i,j) given
        below at the discription of output parameters v and w.

 mdab   the first dimension of the arrays br,bi,cr, and ci as it
        appears in the program that calls vhsec.

 ndab   the second dimension of the arrays br,bi,cr, and ci as it
        appears in the program that calls vhsec.

 wvhsec an array which must be initialized by subroutine vhseci.

 lvhsec the dimension of the array wvhsec as it appears in the
        program that calls vhsec.

 work   a work array that does not have to be saved.

 lwork  the dimension of the array work as it appears in the
        program that calls vhsec.

*************************************************************
```

```
 output parameters

 v,w    two or three dimensional arrays (see input parameter nt)
        in which the synthesis is stored.

 ierror = 0  no errors
        = 1  error in the specification of nlat
        = 2  error in the specification of nlon
        = 3  error in the specification of ityp
        = 4  error in the specification of nt
        = 5  error in the specification of idvw
        = 6  error in the specification of jdvw
        = 7  error in the specification of mdab
        = 8  error in the specification of ndab
        = 9  error in the specification of lvhsec
        = 10 error in the specification of lwork
*****************************************************************
```

vhseci(...)

```
*****************************************************************

 subroutine vhseci initializes the array wvhsec which can then be
 used repeatedly by subroutine vhsec until nlat or nlon is changed

*****************************************************************

 input parameters

 nlat   the number of colatitudes on the full sphere including the
        poles.

 nlon   the number of distinct londitude points.

 lvhsec the dimension of the array wvhsec as it appears in the
        program that calls vhsec.

 dwork  a doubleprecision work array that does not have to be save

 ldwork the dimension of the array dwork as it appears in the
        program that calls vhsec.

*****************************************************************

 output parameters

 wvhsec an array which is initialized for use by subroutine vhsec.

 ierror = 0  no errors
        = 1  error in the specification of nlat
        = 2  error in the specification of nlon
        = 3  error in the specification of lvhsec
```

```
            = 4  error in the specification of ldwork
     ************************************************************
```

vhses(...)

```
     ************************************************************

      subroutine vhses performs the vector spherical harmonic synthesis
      of the arrays br, bi, cr, and ci and stores the result in the
      arrays v and w. v(i,j) and w(i,j) are the colatitudinal
      (measured from the north pole) and east longitudinal components
      respectively, located at colatitude theta(i) = (i-1)*pi/(nlat-1)
      and longitude phi(j) = (j-1)*2*pi/nlon. the spectral
      representation of (v,w) is given below at output parameters v,w.


     ************************************************************

      input parameters

      nlat   the number of colatitudes on the full sphere including the
             poles.

      nlon   the number of distinct longitude points.

      ityp   = 0  no symmetries exist about the equator. the synthesis
                  is performed on the entire sphere.

      nt     the number of syntheses.

      idvw   the first dimension of the arrays v,w as it appears in
             the program that calls vhaes.

      jdvw   the second dimension of the arrays v,w as it appears in
             the program that calls vhses.

      br,bi  two or three dimensional arrays (see input parameter nt)
      cr,ci  that contain the vector spherical harmonic coefficients
             in the spectral representation of v(i,j) and w(i,j) given

      mdab   the first dimension of the arrays br,bi,cr, and ci as it

      ndab   the second dimension of the arrays br,bi,cr, and ci as it


      lvhses the dimension of the array wvhses as it appears in the
             program that calls vhses.

      work   a work array that does not have to be saved.

      lwork  the dimension of the array work as it appears in the
             program that calls vhses.
```

```
*****************************************************************

 output parameters

 v,w    two or three dimensional arrays (see input parameter nt)
        in which the synthesis is stored.

 ierror = 0  no errors
        = 1  error in the specification of nlat
        = 2  error in the specification of nlon
        = 3  error in the specification of ityp
        = 4  error in the specification of nt
        = 5  error in the specification of idvw
        = 6  error in the specification of jdvw
        = 7  error in the specification of mdab
        = 8  error in the specification of ndab
        = 9  error in the specification of lvhses
        = 10 error in the specification of lwork
*****************************************************************
```

vhsesi(...)

```
*****************************************************************


  subroutine vhsesi initializes the array wvhses which can then be
  used repeatedly by subroutine vhses until nlat or nlon is change

*****************************************************************

  input parameters

  nlat   the number of colatitudes on the full sphere including th
         poles.

  nlon   the number of distinct longitude points.

  lvhses the dimension of the array wvhses as it appears in the
         program that calls vhses.

  work   a work array that does not have to be saved.

  lwork  the dimension of the array work as it appears in the
         program that calls vhses.

  dwork  an unsaved doubleprecision work space

  ldwork the length of the array dwork as it appears in the
         program that calls vhsesi.

*****************************************************************
```

```
        output parameters

        wvhses an array which is initialized for use by subroutine vhses

        ierror = 0  no errors
               = 1  error in the specification of nlat
               = 2  error in the specification of nlon
               = 3  error in the specification of lvhses
               = 4  error in the specification of lwork
               = 5  error in the specification of ldwork
      ****************************************************************
```

vhsgc(...)

```
      ****************************************************************


        subroutine vhsgc performs the vector spherical harmonic synthesi
        of the arrays br, bi, cr, and ci and stores the result in the
        arrays v and w. v(i,j) and w(i,j) are the colatitudinal
        (measured from the north pole) and east longitudinal components
        respectively, located at the gaussian colatitude point theta(i)
        and longitude phi(j) = (j-1)*2*pi/nlon. the spectral
        representation of (v,w) is given below at output parameters v,w.


      ************************************************************

        input parameters

        nlat   the number of points in the gaussian colatitude grid on t
               full sphere.

        nlon   the number of distinct longitude points.

        ityp   = 0  no symmetries exist about the equator. the synthesis
                    is performed on the entire sphere.

        nt     the number of syntheses.

        idvw   the first dimension of the arrays v,w as it appears in
               the program that calls vhsgc.

        jdvw   the second dimension of the arrays v,w as it appears in
               the program that calls vhsgc.

        br,bi  two or three dimensional arrays (see input parameter nt)
        cr,ci  that contain the vector spherical harmonic coefficients
               in the spectral representation of v(i,j) and w(i,j) given
               below at the discription of output parameters v and w.

        mdab   the first dimension of the arrays br,bi,cr, and ci as it
               appears in the program that calls vhsgc.
```

```
        ndab   the second dimension of the arrays br,bi,cr, and ci as it
               appears in the program that calls vhsgc.

        wvhsgc an array which must be initialized by subroutine vhsgci.

        lvhsgc the dimension of the array wvhsgc as it appears in the
               program that calls vhsgc.

        work   a work array that does not have to be saved.
               program that calls vhsgc.
        lwork  the dimension of the array work as it appears in the


      ****************************************************************

        output parameters

        v,w    two or three dimensional arrays (see input parameter nt)
               in which the synthesis is stored.

        ierror = 0  no errors
               = 1  error in the specification of nlat
               = 2  error in the specification of nlon
               = 3  error in the specification of ityp
               = 4  error in the specification of nt
               = 5  error in the specification of idvw
               = 6  error in the specification of jdvw
               = 7  error in the specification of mdab
               = 8  error in the specification of ndab
               = 9  error in the specification of lvhsgc
               = 10 error in the specification of lwork
      ****************************************************************
```

vhsgci(...)

```
        subroutine vhsgci initializes the array wvhsgc which can then be
        used repeatedly by subroutine vhsgc until nlat or nlon is change

      ****************************************************************

        input parameters

        nlat   the number of points in the gaussian colatitude grid on t
               full sphere.

        nlon   the number of distinct longitude points.

        lvhsgc the dimension of the array wvhsgc as it appears in the
               program that calls vhsgc.

        ldwork the dimension of the array dwork as it appears in the
               program that calls vhsgsi.
```

```
      ************************************************************

        output parameters

        wvhsgc an array which is initialized for use by subroutine vhsgc

        ierror = 0  no errors
               = 1  error in the specification of nlat
               = 2  error in the specification of nlon
               = 3  error in the specification of lvhsgc
               = 4  error in the specification of ldwork
      ************************************************************
```

vhsgs(...)

```
        subroutine vhsgci initializes the array wvhsgc which can then be

       subroutine vhsgs performs the vector spherical harmonic synthesis
       of the arrays br, bi, cr, and ci and stores the result in the
       arrays v and w.  the synthesis is performed on an equally spaced
       longitude grid and a gaussian colatitude grid (measured from
       the north pole). v(i,j) and w(i,j) are the colatitudinal and
       east longitudinal components respectively, located at the i(th)
       colatitude gaussian point (see nlat below) and longitude
       phi(j) = (j-1)*2*pi/nlon.  the spectral respresentation of (v,w)
       is given below at output parameters v,w.


      ************************************************************

       input parameters

       nlat   the number of points in the gaussian colatitude grid on th
              full sphere.

       nlon   the number of distinct longitude points.

       ityp   = 0  no symmetries exist about the equator. the synthesis
                   is performed on the entire sphere.

       nt     the number of syntheses.

       idvw   the first dimension of the arrays v,w as it appears in
              the program that calls vhags.

       jdvw   the second dimension of the arrays v,w as it appears in
              the program that calls vhsgs.

       br,bi  two or three dimensional arrays (see input parameter nt)
       cr,ci  that contain the vector spherical harmonic coefficients
              in the spectral representation of v(i,j) and w(i,j) given
              below at the discription of output parameters v and w.
```

```
      mdab   the first dimension of the arrays br,bi,cr, and ci as it
             appears in the program that calls vhsgs.

      ndab   the second dimension of the arrays br,bi,cr, and ci as it
             appears in the program that calls vhsgs.

      wvhsgs an array which must be initialized by subroutine vhsgsi.

      lvhsgs the dimension of the array wvhsgs as it appears in the
             program that calls vhsgs.

      work   a work array that does not have to be saved.

      lwork  the dimension of the array work as it appears in the
             program that calls vhsgs.


     ****************************************************************

      output parameters

      v,w    two or three dimensional arrays (see input parameter nt)
             in which the synthesis is stored.

      ierror = 0  no errors
             = 1  error in the specification of nlat
             = 2  error in the specification of nlon
             = 3  error in the specification of ityp
             = 4  error in the specification of nt
             = 5  error in the specification of idvw
             = 6  error in the specification of jdvw
             = 7  error in the specification of mdab
             = 8  error in the specification of ndab
             = 9  error in the specification of lvhsgs
             = 10 error in the specification of lwork
     ****************************************************************
```

vhsgsi(...)

```
       subroutine vhsgci initializes the array wvhsgc which can then be
      subroutine vhsgsi initializes the array wvhsgs which can then be
      used repeatedly by subroutine vhsgs until nlat or nlon is changed


     ****************************************************************

      input parameters

      nlat   the number of points in the gaussian colatitude grid on th
             full sphere.
      nlon   the number of distinct longitude points.
```

```
      lvhsgs the dimension of the array wvhsgs as it appears in the
             program that calls vhsgs.

      dwork a doubleprecision work array that does not need to be saved

      ldwork the dimension of the array dwork as it appears in the
             program that calls vhsgsi.


     ************************************************************

      output parameters

      wvhsgs an array which is initialized for use by subroutine vhsgs.

      ierror = 0  no errors
             = 1  error in the specification of nlat
             = 2  error in the specification of nlon
             = 3  error in the specification of lvhsgs
             = 4  error in the specification of lwork
     ************************************************************
```

vlapec(...)

```
      subroutine vhsgci initializes the array wvhsgc which can then be
      subroutine vlapec computes the vector laplacian of the vector fie
      (v,w) in (vlap,wlap) (see the definition of the vector laplacian
      the output parameter description of vlap,wlap below).  w and wlap
      are east longitudinal components of the vectors.  v and vlap are
      colatitudinal components of the vectors.  br,bi,cr, and ci are th
      vector harmonic coefficients of (v,w).  these must be precomputed
      vhaec and are input parameters to vlapec.  the laplacian componen
      in (vlap,wlap) have the same symmetry or lack of symmetry about t
      equator as (v,w).  the input parameters ityp,nt,mdbc,nbdc must ha
      the same values used by vhaec to compute br,bi,cr, and ci for (v,
      vlap(i,j) and wlap(i,j) are given on the sphere at the colatitude

             theta(i) = (i-1)*pi/(nlat-1)

      for i=1,...,nlat and east longitude

             lambda(j) = (j-1)*2*pi/nlon

      for j=1,...,nlon.

     ************************************************************

      input parameters

      nlat   the number of colatitudes on the full sphere including the
             poles.

      nlon   the number of distinct longitude points.
```

```
 ityp   this parameter should have the same value input to subrout
        vhaec to compute the coefficients br,bi,cr, and ci for the
        vector field (v,w).  ityp is set as follows:

        = 0  no symmetries exist in (v,w) about the equator. (vlap
             is computed and stored on the entire sphere.

 nt     nt is the number of vector fields (v,w).

 idvw   the first dimension of the arrays vlap and wlap as it appe
        in the program that calls vlapec.

 jdvw   the second dimension of the arrays vlap and wlap as it app
        in the program that calls vlapec.


 br,bi  two or three dimensional arrays (see input parameter nt)
 cr,ci  that contain vector spherical harmonic coefficients
        of the vector field (v,w) as computed by subroutine vhaec.
        br,bi,cr and ci must be computed by vhaec prior to calling
        vlapec.

mdbc    the first dimension of the arrays br,bi,cr and ci as it
        appears in the program that calls vlapec.

ndbc    the second dimension of the arrays br,bi,cr and ci as it
        appears in the program that calls vlapec.

wvhsec  an array which must be initialized by subroutine vhseci.
        of vlapec.

lvhsec  the dimension of the array wvhsec as it appears in the
        program that calls vlapec.

work    a work array that does not have to be saved.

lwork   the dimension of the array work as it appears in the
        program that calls vlapec.

**************************************************************

 output parameters


vlap,   two or three dimensional arrays (see input parameter nt) t
wlap    contain the vector laplacian of the field (v,w).

ierror    a parameter which flags errors in input parameters as fo
        = 0  no errors detected
        = 1  error in the specification of nlat
        = 2  error in the specification of nlon
```

```
            = 3  error in the specification of ityp
            = 4  error in the specification of nt
            = 5  error in the specification of idvw
            = 6  error in the specification of jdvw
            = 7  error in the specification of mdbc
            = 8  error in the specification of ndbc
            = 9  error in the specification of lvhsec
            = 10 error in the specification of lwork (lwork < lwkmin)
     ****************************************************************
```

vlapes(...)

```
      subroutine vlapes computes the vector laplacian of the vector fie
      (v,w) in (vlap,wlap) (see the definition of the vector laplacian
      the output parameter description of vlap,wlap below).  w and wlap
      are east longitudinal components of the vectors.  v and vlap are
      colatitudinal components of the vectors.  br,bi,cr, and ci are th
      vector harmonic coefficients of (v,w).  these must be precomputed
      vhaes and are input parameters to vlapes.  the laplacian componen
      in (vlap,wlap) have the same symmetry or lack of symmetry about t
      equator as (v,w).  the input parameters ityp,nt,mdbc,nbdc must ha
      the same values used by vhaes to compute br,bi,cr, and ci for (v,
      vlap(i,j) and wlap(i,j) are given on the sphere at the colatitude

             theta(i) = (i-1)*pi/(nlat-1)

      for i=1,...,nlat and east longitude

             lambda(j) = (j-1)*2*pi/nlon

      for j=1,...,nlon.

     ****************************************************************

      input parameters

      nlat   the number of colatitudes on the full sphere including the
             poles.

      nlon   the number of distinct longitude points.

      ityp   this parameter should have the same value input to subrout
             vhaes to compute the coefficients br,bi,cr, and ci for the
             vector field (v,w).  ityp is set as follows:

             = 0  no symmetries exist in (v,w) about the equator. (vlap
                  is computed and stored on the entire sphere.

      nt     nt is the number of vector fields (v,w).

     idvw     the first dimension of the arrays vlap and wlap as it app
             in the program that calls vlapes.
```

```
      jdvw     the second dimension of the arrays vlap and wlap as it ap
              in the program that calls vlapes.

      br,bi    two or three dimensional arrays (see input parameter nt)
      cr,ci    that contain vector spherical harmonic coefficients
              of the vector field (v,w) as computed by subroutine vhaes.
              br,bi,cr and ci must be computed by vhaes prior to calling
              vlapes.

      mdbc     the first dimension of the arrays br,bi,cr and ci as it
              appears in the program that calls vlapes.

      ndbc     the second dimension of the arrays br,bi,cr and ci as it
              appears in the program that calls vlapes.

      wvhses   an array which must be initialized by subroutine vhsesi.

      lvhses   the dimension of the array wvhses as it appears in the
              program that calls vlapes.

       work    a work array that does not have to be saved.

       lwork   the dimension of the array work as it appears in the
              program that calls vlapes.

      ************************************************************

       output parameters

      vlap,    two or three dimensional arrays (see input parameter nt) t
      wlap     contain the vector laplacian of the field (v,w).

      ierror     a parameter which flags errors in input parameters as fo
              = 0  no errors detected
              = 1  error in the specification of nlat
              = 2  error in the specification of nlon
              = 3  error in the specification of ityp
              = 4  error in the specification of nt
              = 5  error in the specification of idvw
              = 6  error in the specification of jdvw
              = 7  error in the specification of mdbc
              = 8  error in the specification of ndbc
              = 9  error in the specification of lvhses
              = 10 error in the specification of lwork
      ************************************************************
```

vlapgc(...)

```
       subroutine vlapes computes the vector laplacian of the vector fie
       given the vector spherical harmonic coefficients (br,bi,cr,ci)
       precomputed by subroutine vhagc for a vector field (v,w), subrout
       vlapgc computes the vector laplacian of the vector field (v,w)
       in (vlap,wlap) (see the definition of the vector laplacian at
```

```
 the output parameter description of vlap,wlap below).  w and wlap
 are east longitudinal components of the vectors.  v and vlap are
 colatitudinal components of the vectors.  the laplacian component
 in (vlap,wlap) have the same symmetry or lack of symmetry about t
 equator as (v,w).  the input parameters ityp,nt,mdbc,nbdc must ha
 the same values used by vhagc to compute br,bi,cr, and ci for (v,
 vlap(i,j) and wlap(i,j) are given on the sphere at the gaussian
 colatitude theta(i) (see nlat as input parameter) and east longit
 lambda(j) = (j-1)*2*pi/nlon for i=1,...,nlat and j=1,...,nlon.


**************************************************************

 input parameters

 nlat   the number of points in the gaussian colatitude grid on th
        full sphere.

 nlon   the number of distinct longitude points.

 ityp   this parameter should have the same value input to subrout
        vhagc to compute the coefficients br,bi,cr, and ci for the
        vector field (v,w).  ityp is set as follows:

        = 0  no symmetries exist in (v,w) about the equator. (vlap
             is computed and stored on the entire sphere.

 nt     nt is the number of vector fields (v,w).

idvw    the first dimension of the arrays vlap and wlap as it appe
        in the program that calls vlapgc.

jdvw    the second dimension of the arrays vlap and wlap as it app
        in the program that calls vlapgc.

br,bi   two or three dimensional arrays (see input parameter nt)
cr,ci   that contain vector spherical harmonic coefficients
        of the vector field (v,w) as computed by subroutine vhagc.
        br,bi,cr and ci must be computed by vhagc prior to calling
        vlapgc.

mdbc    the first dimension of the arrays br,bi,cr and ci as it
        appears in the program that calls vlapgc.

ndbc    the second dimension of the arrays br,bi,cr and ci as it
        appears in the program that calls vlapgc.

wvhsgc  an array which must be initialized by subroutine vhsgci.

lvhsgc  the dimension of the array wvhsgc as it appears in the
        program that calls vhagc.
```

```
 work   a work array that does not have to be saved.

 lwork  the dimension of the array work as it appears in the
        program that calls vlapgc.

***************************************************************

 output parameters


vlap,   two or three dimensional arrays (see input parameter nt) t
wlap    contain the vector laplacian of the field (v,w).

ierror   a parameter which flags errors in input parameters as fol
        = 0  no errors detected
        = 1  error in the specification of nlat
        = 2  error in the specification of nlon
        = 3  error in the specification of ityp
        = 4  error in the specification of nt
        = 5  error in the specification of idvw
        = 6  error in the specification of jdvw
        = 7  error in the specification of mdbc
        = 8  error in the specification of ndbc
        = 9  error in the specification of lvhsgc
        = 10 error in the specification of lwork
***************************************************************
```

***vlapgs*(...)**

```
 subroutine vlapes computes the vector laplacian of the vector fie
  given the vector spherical harmonic coefficients (br,bi,cr,ci)
  precomputed by subroutine vhags for a vector field (v,w), subrou
  vlapgs computes the vector laplacian of the vector field (v,w)
  in (vlap,wlap) (see the definition of the vector laplacian at
  the output parameter description of vlap,wlap below).  w and wla
  are east longitudinal components of the vectors.  v and vlap are
  colatitudinal components of the vectors.  the laplacian componen
  in (vlap,wlap) have the same symmetry or lack of symmetry about
  equator as (v,w).  the input parameters ityp,nt,mdbc,nbdc must h
  the same values used by vhags to compute br,bi,cr, and ci for (v
  vlap(i,j) and wlap(i,j) are given on the sphere at the gaussian
  colatitude theta(i) (see nlat as input parameter) and east longi
  lambda(j) = (j-1)*2*pi/nlon for i=1,...,nlat and j=1,...,nlon.


***************************************************************

  input parameters

  nlat   the number of points in the gaussian colatitude grid on t
         full sphere.

  nlon   the number of distinct longitude points.
```

```
  ityp   this parameter should have the same value input to subrou
         vhags to compute the coefficients br,bi,cr, and ci for th
         vector field (v,w).  ityp is set as follows:

         = 0  no symmetries exist in (v,w) about the equator. (vla
              is computed and stored on the entire sphere.

  nt     nt is the number of vector fields (v,w).

  idvw   the first dimension of the arrays vlap and wlap as it app
         in the program that calls vlapgs.

  jdvw   the second dimension of the arrays vlap and wlap as it ap
         in the program that calls vlapgs.


  br,bi  two or three dimensional arrays (see input parameter nt)
  cr,ci  that contain vector spherical harmonic coefficients
         of the vector field (v,w) as computed by subroutine vhags
         br,bi,cr and ci must be computed by vhags prior to callin
         vlapgs.

  mdbc   the first dimension of the arrays br,bi,cr and ci as it
         appears in the program that calls vlapgs.

  ndbc   the second dimension of the arrays br,bi,cr and ci as it
         appears in the program that calls vlapgs. ndbc must be at
         least nlat.

  wvhsgs an array which must be initialized by subroutine vlapgsi

  lvhsgs the dimension of the array wvhsgs as it appears in the
         program that calls vlapgs.

  work   a work array that does not have to be saved.

  lwork  the dimension of the array work as it appears in the
         program that calls vlapgs.

**************************************************************

  output parameters


 vlap,   two or three dimensional arrays (see input parameter nt)
 wlap    contain the vector laplacian of the field (v,w).

 ierror  a parameter which flags errors in input parameters as fol

         = 0  no errors detected
         = 1  error in the specification of nlat
```

```
            = 2  error in the specification of nlon
            = 3  error in the specification of ityp
            = 4  error in the specification of nt
            = 5  error in the specification of idvw
            = 6  error in the specification of jdvw
            = 7  error in the specification of mdbc
            = 8  error in the specification of ndbc
            = 9  error in the specification of lvhsgs
            = 10 error in the specification of lwork
     ************************************************************
```

vrtec(...)

```
      subroutine vlapes computes the vector laplacian of the vector fie
       given the vector spherical harmonic coefficients cr and ci, prec
       by subroutine vhaec for a vector field (v,w), subroutine vrtec
       computes the vorticity of the vector field in the scalar array
       vt.  vt(i,j) is the vorticity at the colatitude

              theta(i) = (i-1)*pi/(nlat-1)

       and longitude

              lambda(j) = (j-1)*2*pi/nlon

       on the sphere.  i.e.,

              vt(i,j) =  [-dv/dlambda + d(sint*w)/dtheta]/sint

       where sint = sin(theta(i)).  w is the east longitudinal and v
       is the colatitudinal component of the vector field from which
       cr,ci were precomputed.  required associated legendre polynomial
       are recomputed rather than stored as they are in subroutine vrte

     ************************************************************

       input parameters

       nlat   the number of colatitudes on the full sphere including th
              poles.

       nlon   the number of distinct longitude points.

       isym   a parameter which determines whether the vorticity is
              computed on the full or half sphere as follows:
        = 0
              the symmetries/antsymmetries described in isym=1,2 below
              do not exist in (v,w) about the equator.

       ivrt   the first dimension of the array vt as it appears in
              the program that calls vrtec.

       jvrt   the second dimension of the array vt as it appears in
```

```
            the program that calls vrtec.

      cr,ci  two or three dimensional arrays (see input parameter nt)
             that contain vector spherical harmonic coefficients
             of the vector field (v,w) as computed by subroutine vhaec
             cr and ci must be computed by vhaec prior to calling
             vrtec.

      mdc    the first dimension of the arrays cr and ci as it
             appears in the program that calls vrtec.

      ndc    the second dimension of the arrays cr and ci as it
             appears in the program that calls vrtec.

      wshsec an array which must be initialized by subroutine shseci.

      lshsec the dimension of the array wshsec as it appears in the
             program that calls vrtec.

      work   a work array that does not have to be saved.

     lwork   the dimension of the array work as it appears in the
             program that calls vrtec.

    ************************************************************

      output parameters

      vort   a two or three dimensional array (see input parameter nt)
             that contains the vorticity of the vector field (v,w)
             whose coefficients cr,ci where computed by subroutine vha

    ierror   an error parameter which indicates fatal errors with inpu
             parameters when returned positive.
           = 0  no errors
           = 1  error in the specification of nlat
           = 2  error in the specification of nlon
           = 3  error in the specification of isym
           = 4  error in the specification of nt
           = 5  error in the specification of ivrt
           = 6  error in the specification of jvrt
           = 7  error in the specification of mdc
           = 8  error in the specification of ndc
           = 9  error in the specification of lshsec
           = 10 error in the specification of lwork
    ************************************************************
```

vrtes(...)

```
     subroutine vlapes computes the vector laplacian of the vector fie
      given the vector spherical harmonic coefficients cr and ci, prec
      by subroutine vhaes for a vector field (v,w), subroutine vrtes
      computes the vorticity of the vector field in the scalar array
```

```
     vt.  vt(i,j) is the vorticity at the colatitude

            theta(i) = (i-1)*pi/(nlat-1)

     and longitude

            lambda(j) = (j-1)*2*pi/nlon

     on the sphere.  i.e.,

            vt(i,j) =  [-dv/dlambda + d(sint*w)/dtheta]/sint

     where sint = sin(theta(i)).  w is the east longitudinal and v
     is the colatitudinal component of the vector field from which
     cr,ci were precomputed.  required associated legendre polynomial
     are stored rather than recomputed  as they are in subroutine vrt

    ************************************************************

     input parameters

     nlat   the number of colatitudes on the full sphere including th
            poles.

     nlon   the number of distinct londitude points.

     isym   a parameter which determines whether the vorticity is
            computed on the full or half sphere as follows:
      = 0
            the symmetries/antsymmetries described in isym=1,2 below
            do not exist in (v,w) about the equator. the vorticity is
            computed on the entire sphere.

      nt    nt is the number of scalar and vector fields.

     ivrt   the first dimension of the array vt as it appears in
            the program that calls vrtes.

     jvrt   the second dimension of the array vt as it appears in
            the program that calls vrtes.

     cr,ci  two or three dimensional arrays (see input parameter nt)
            that contain vector spherical harmonic coefficients
            of the vector field (v,w) as computed by subroutine vhaes
            cr and ci must be computed by vhaes prior to calling
            vrtes.

     mdc    the first dimension of the arrays cr and ci as it
            appears in the program that calls vrtes.

     ndc    the second dimension of the arrays cr and ci as it
            appears in the program that calls vrtes.
```

```
 wshses  an array which must be initialized by subroutine shsesi.

 lshses  the dimension of the array wshses as it appears in the
         program that calls vrtes.

 work    a work array that does not have to be saved.

 lwork   the dimension of the array work as it appears in the
         program that calls vrtes.

*****************************************************************

  output parameters

  vort   a two or three dimensional array (see input parameter nt)
         that contains the vorticity of the vector field (v,w)
         whose coefficients cr,ci where computed by subroutine vha

 ierror  an error parameter which indicates fatal errors with inpu
         parameters when returned positive.
       = 0  no errors
       = 1  error in the specification of nlat
       = 2  error in the specification of nlon
       = 3  error in the specification of isym
       = 4  error in the specification of nt
       = 5  error in the specification of ivrt
       = 6  error in the specification of jvrt
       = 7  error in the specification of mdc
       = 8  error in the specification of ndc
       = 9  error in the specification of lshses
       = 10 error in the specification of lwork
*****************************************************************
```

vrtgc(...)

```
 subroutine vlapes computes the vector laplacian of the vector fie
  given the vector spherical harmonic coefficients cr and ci, prec
  by subroutine vhagc for a vector field (v,w), subroutine vrtgc
  computes the vorticity of the vector field in the scalar array
  vort.  vort(i,j) is the vorticity at the gaussian colatitude
  theta(i) (see nlat as input parameter) and longitude
  lambda(j) = (j-1)*2*pi/nlon on the sphere.  i.e.,

         vort(i,j) =  [-dv/dlambda + d(sint*w)/dtheta]/sint

  where sint = sin(theta(i)).  w is the east longitudinal and v
  is the colatitudinal component of the vector field from which
  cr,ci were precomputed.  required associated legendre polynomial
  are recomputed rather than stored as they are in subroutine vrtg


*****************************************************************
```

```
 input parameters

 nlat   the number of points in the gaussian colatitude grid on t
        full sphere.

 nlon   the number of distinct longitude points.

 isym   a parameter which determines whether the vorticity is
        computed on the full or half sphere as follows:
  = 0
        the symmetries/antsymmetries described in isym=1,2 below
        do not exist in (v,w) about the equator. the vorticity is
        computed on the entire sphere.

  nt    nt is the number of scalar and vector fields.

 ivrt   the first dimension of the array vort as it appears in
        the program that calls vrtgc.

 jvrt   the second dimension of the array vort as it appears in
        the program that calls vrtgc.

cr,ci   two or three dimensional arrays (see input parameter nt)
        that contain vector spherical harmonic coefficients
        of the vector field (v,w) as computed by subroutine vhagc
        cr and ci must be computed by vhagc prior to calling
        vrtgc.

  mdc   the first dimension of the arrays cr and ci as it
        appears in the program that calls vrtgc.

  ndc   the second dimension of the arrays cr and ci as it
        appears in the program that calls vrtgc.

wshsgc  an array which must be initialized by subroutine shsgci.

lshsgc  the dimension of the array wshsgc   as it appears in the
        program that calls vrtgc.

 work   a work array that does not have to be saved.

lwork   the dimension of the array work as it appears in the
        program that calls vrtgc.


****************************************************************

 output parameters


 vort   a two or three dimensional array (see input parameter nt)
```

```
              that contains the vorticity of the vector field (v,w)
              whose coefficients cr,ci where computed by subroutine vha

      ierror  an error parameter which indicates fatal errors with inpu
              parameters when returned positive.
            = 0  no errors
            = 1  error in the specification of nlat
            = 2  error in the specification of nlon
            = 3  error in the specification of isym
            = 4  error in the specification of nt
            = 5  error in the specification of ivrt
            = 6  error in the specification of jvrt
            = 7  error in the specification of mdc
            = 8  error in the specification of ndc
            = 9  error in the specification of lshsgc
            = 10 error in the specification of lwork
      ************************************************************
```

vrtgs(...)

```
      ************************************************************

      given the vector spherical harmonic coefficients cr and ci, precom
      by subroutine vhags for a vector field (v,w), subroutine vrtgs
      computes the vorticity of the vector field in the scalar array
      vort.  vort(i,j) is the vorticity at the gaussian colatitude
      theta(i) (see nlat as input parameter) and longitude
      lambda(j) = (j-1)*2*pi/nlon on the sphere.  i.e.,

             vort(i,j) =  [-dv/dlambda + d(sint*w)/dtheta]/sint

      where sint = sin(theta(i)).  w is the east longitudinal and v
      is the colatitudinal component of the vector field from which
      cr,ci were precomputed.  required associated legendre polynomials
      are stored rather than recomputed as they are in subroutine vrtgc.


      ************************************************************

      input parameters

      nlat   the number of points in the gaussian colatitude grid on the

      nlon   the number of distinct longitude points.

      isym   a parameter which determines whether the vorticity is
             computed on the full or half sphere as follows:

       = 0
             the symmetries/antsymmetries described in isym=1,2 below
             do not exist in (v,w) about the equator.

       nt    nt is the number of scalar and vector fields.
```

```
ivrt   the first dimension of the array vort as it appears in
       the program that calls vrtgs.

jvrt   the second dimension of the array vort as it appears in
       the program that calls vrtgs.

cr,ci   two or three dimensional arrays (see input parameter nt)
       that contain vector spherical harmonic coefficients
       of the vector field (v,w) as computed by subroutine vhags.
       cr and ci must be computed by vhags prior to calling
       vrtgs.

 mdc   the first dimension of the arrays cr and ci as it
       appears in the program that calls vrtgs.

 ndc   the second dimension of the arrays cr and ci as it
       appears in the program that calls vrtgs.

wshsgs an array which must be initialized by subroutine shsgsi.

lshsgs the dimension of the array wshsgs   as it appears in the
       program that calls vrtgs.

work   a work array that does not have to be saved.

lwork  the dimension of the array work as it appears in the
       program that calls vrtgs.

************************************************************


output parameters

vort   a two or three dimensional array (see input parameter nt)
       that contains the vorticity of the vector field (v,w)
       whose coefficients cr,ci where computed by subroutine vhags

ierror= 0  no errors
     = 0  no errors
     = 1  error in the specification of nlat
     = 2  error in the specification of nlon
     = 3  error in the specification of isym
     = 4  error in the specification of nt
     = 5  error in the specification of ivrt
     = 6  error in the specification of jvrt
     = 7  error in the specification of mdc
     = 8  error in the specification of ndc
     = 9  error in the specification of lshsgs
     = 10 error in the specification of lwork
************************************************************
```

vshifte(...)

```
      ****************************************************************
        subroutine vshifte does a highly accurate 1/2 grid increment shi
        in both longitude and latitude of equally spaced vector data on
        sphere. data is transferred between the nlon by nlat offset grid
        in (uoff,voff) (which excludes poles) and the nlon by nlat+1 reg
        grid in (ureg,vreg) (which includes poles).  the transfer can go
        (uoff,voff) to (ureg,vreg) or vice versa (see ioff).  the grids
        underly the vector fields are described below.  the north and so
        pole are at 0.5*pi and-0.5*pi radians respectively (pi=4.*atan(1
        uoff and ureg are the east longitudinal vector data components.
        and vreg are the latitudinal vector data components.

        subroutine vshifte here does a shift to the offset grid

      ************************************************************

        input parameters

        nlon    the number of longitude points on both the offset and re
                uniform grid in longitude.

        nlat    the number of latitude points on the offset uniform grid
                is the number of latitude points on the regular uniform

        ureg    a nlon by nlat+1 array that contains the east longitudin
                data component on the regular grid described above.

        vreg    a nlon by nlat+1 array that contains the latitudinal vec
                component on the regular grid described above.

        uoff    a nlon by nlat array that contains the east longitudinal
                data component on the offset grid described above.

        voff    a nlon by nlat array that contains the latitudinal vecto
                component on the offset grid described above.


        wsav    a real saved work space array that must be initialized b
                subroutine vshiftei(nlon,nlat,wsav,ier) before calling v

        lsav    the length of the saved work space wsav in the routine c
                and sshiftei.

        wrk     a real unsaved work space

        lwrk    the length of the unsaved work space in the routine call

      ************************************************************

        output parameters
        ier = 0 if no errors are detected
```

```
           = 1 if ioff is not equal to 0 or 1
           = 2 if nlon < 4
           = 3 if nlat < 3
           = 4 if lsave < 2*(nlon+2*nlat)+32
           = 5 if lwork < 2*nlon*(nlat+1) for nlon even or
              lwork < nlon*(5*nlat+1) for nlon odd
     ****************************************************************
```

vshifti(...)

```
       subroutine vshifti initializes the saved work space wsav
       for ioff and nlon and nlat (see documentation for vshifte).
       vshifti must be called before vshifte whenever ioff or nlon
       or nlat change.

       ier = 0 if no errors with input arguments
           = 1 if ioff is not 0 or 1
           = 2 if nlon < 4
           = 3 if nlat < 3
           = 4 if lsav < 2*(2*nlat+nlon+16)
     ****************************************************************
```

vtsec(...)

```
     ****************************************************************

     given the vector harmonic analysis br,bi,cr, and ci (computed
     by subroutine vhaec) of some vector function (v,w), this
     subroutine computes the vector function (vt,wt) which is
     the derivative of (v,w) with respect to colatitude theta. vtsec
     is similar to vhsec except the vector harmonics are replaced by
     their derivative with respect to colatitude with the result that
     (vt,wt) is computed instead of (v,w). vt(i,j) is the derivative
     of the colatitudinal component v(i,j) at the point theta(i) =
     (i-1)*pi/(nlat-1) and longitude phi(j) = (j-1)*2*pi/nlon. the
     spectral representation of (vt,wt) is given below at output
     parameters vt,wt.

     ************************************************************

     input parameters

     nlat   the number of colatitudes on the full sphere including the
            poles.

     nlon   the number of distinct londitude points.

     ityp   = 0  no symmetries exist about the equator. the synthesis
                 is performed on the entire sphere.

     nt     the number of syntheses.  in the program that calls vtsec,
            the arrays vt,wt,br,bi,cr, and ci can be three dimensional
            in which case multiple syntheses will be performed.
```

```
      idvw   the first dimension of the arrays vt,wt as it appears in
             the program that calls vtsec.

      jdvw   the second dimension of the arrays vt,wt as it appears in
             the program that calls vtsec. jdvw must be at least nlon.

      br,bi  two or three dimensional arrays (see input parameter nt)
      cr,ci  that contain the vector spherical harmonic coefficients
             of (v,w) as computed by subroutine vhaec.

      mdab   the first dimension of the arrays br,bi,cr, and ci as it
             appears in the program that calls vtsec.

      ndab   the second dimension of the arrays br,bi,cr, and ci as it
             appears in the program that calls vtsec.

      wvts   an array which must be initialized by subroutine vtseci.

      lwvts  the dimension of the array wvts as it appears in the
             program that calls vtsec.

      work   a work array that does not have to be saved.

      lwork  the dimension of the array work as it appears in the
             program that calls vtsec.

      ****************************************************************


      output parameters

      vt,wt  two or three dimensional arrays (see input parameter nt)
             in which the derivative of (v,w) with respect to
             colatitude theta is stored.

      ierror = 0  no errors
             = 1  error in the specification of nlat
             = 2  error in the specification of nlon
             = 3  error in the specification of ityp
             = 4  error in the specification of nt
             = 5  error in the specification of idvw
             = 6  error in the specification of jdvw
             = 7  error in the specification of mdab
             = 8  error in the specification of ndab
             = 9  error in the specification of lwvts
             = 10 error in the specification of lwork
      ****************************************************************
```

vtseci(...)

```
      ****************************************************************

        subroutine vtseci initializes the array wvts which can then be
```

```
  used repeatedly by subroutine vtsec until nlat or nlon is change

*****************************************************************

  input parameters

  nlat   the number of colatitudes on the full sphere including th
         poles.

  nlon   the number of distinct longitude points.

  lwvts  the dimension of the array wvts as it appears in the
         program that calls vtsec.

  dwork  a doubleprecision work array that does not have to be sav

  ldwork the dimension of the array work as it appears in the
         program that calls vtsec.

*****************************************************************

  output parameters

  wvts   an array which is initialized for use by subroutine vtsec
         once initialized, wvts can be used repeatedly by vtsec
         as long as nlat or nlon remain unchanged.  wvts must not
         be altered between calls of vtsec.

  ierror = 0  no errors
         = 1  error in the specification of nlat
         = 2  error in the specification of nlon
         = 3  error in the specification of lwvts
         = 4  error in the specification of ldwork
*****************************************************************
```

vtses(...)

```
*****************************************************************

given the vector harmonic analysis br,bi,cr, and ci (computed
by subroutine vhaes) of some vector function (v,w), this
subroutine computes the vector function (vt,wt) which is
the derivative of (v,w) with respect to colatitude theta. vtses
is similar to vhses except the vector harmonics are replaced by
their derivative with respect to colatitude with the result that
(vt,wt) is computed instead of (v,w). vt(i,j) is the derivative
of the colatitudinal component v(i,j) at the point theta(i) =
(i-1)*pi/(nlat-1) and longitude phi(j) = (j-1)*2*pi/nlon. the
spectral representation of (vt,wt) is given below at output
parameters vt,wt.

*****************************************************************
```

```
input parameters

nlat   the number of colatitudes on the full sphere including the
       poles.

nlon   the number of distinct longitude points.

ityp   = 0  no symmetries exist about the equator. the synthesis
            is performed on the entire sphere.

nt     the number of syntheses.

idvw   the first dimension of the arrays vt,wt as it appears in
       the program that calls vtses.

jdvw   the second dimension of the arrays vt,wt as it appears in
       the program that calls vtses.

br,bi  two or three dimensional arrays (see input parameter nt)
cr,ci  that contain the vector spherical harmonic coefficients
       of (v,w) as computed by subroutine vhaes.

mdab   the first dimension of the arrays br,bi,cr, and ci as it
       appears in the program that calls vtses.

ndab   the second dimension of the arrays br,bi,cr, and ci as it
       appears in the program that calls vtses.

wvts   an array which must be initialized by subroutine vtsesi.

lwvts  the dimension of the array wvts as it appears in the
       program that calls vtses.

work   a work array that does not have to be saved.

lwork  the dimension of the array work as it appears in the
       program that calls vtses.


**********************************************************

output parameters

vt,wt  two or three dimensional arrays (see input parameter nt)
       in which the derivative of (v,w) with respect to
       colatitude theta is stored.

ierror = 0  no errors
       = 1  error in the specification of nlat
       = 2  error in the specification of nlon
       = 3  error in the specification of ityp
       = 4  error in the specification of nt
```

```
            = 5  error in the specification of idvw
            = 6  error in the specification of jdvw
            = 7  error in the specification of mdab
            = 8  error in the specification of ndab
            = 9  error in the specification of lwvts
            = 10 error in the specification of lwork
      ************************************************************
```

vtsesi(...)

```
      ************************************************************

        subroutine vtsesi initializes the array wvts which can then be
        used repeatedly by subroutine vtses until nlat or nlon is change

      ************************************************************

        input parameters

        nlat   the number of colatitudes on the full sphere including th
               poles.

        nlon   the number of distinct longitude points.

        lwvts  the dimension of the array wvts as it appears in the
               program that calls vtses.

        work   a work array that does not have to be saved.

        lwork  the dimension of the array work as it appears in the
               program that calls vtses.

        dwork  a doubleprecision work array that does have to be saved.

        ldwork the length of dwork.

      ************************************************************

        output parameters

        wvts   an array which is initialized for use by subroutine vtses

        ierror = 0  no errors
               = 1  error in the specification of nlat
               = 2  error in the specification of nlon
               = 3  error in the specification of lwvts
               = 4  error in the specification of lwork
               = 5  error in the specification of ldwork
      ************************************************************
```

vtsgc(...)

```
      ************************************************************
```

```
given the vector harmonic analysis br,bi,cr, and ci (computed
by subroutine vhagc) of some vector function (v,w), this
subroutine computes the vector function (vt,wt) which is
the derivative of (v,w) with respect to colatitude theta. vtsgc
is similar to vhsgc except the vector harmonics are replaced by
their derivative with respect to colatitude with the result that
(vt,wt) is computed instead of (v,w). vt(i,j) is the derivative
of the colatitudinal component v(i,j) at the gaussian colatitude
theta(i) and longitude phi(j) = (j-1)*2*pi/nlon. the spectral
representation of (vt,wt) is given below at the definition of
output parameters vt,wt.

************************************************************

input parameters

nlat   the number of gaussian colatitudinal grid points theta(i)

nlon   the number of distinct longitude points.

ityp   = 0  no symmetries exist about the equator. the synthesis
            is performed on the entire sphere.

nt     the number of syntheses.

idvw   the first dimension of the arrays vt,wt as it appears in
       the program that calls vtsgc.

jdvw   the second dimension of the arrays vt,wt as it appears in
       the program that calls vtsgc.

br,bi  two or three dimensional arrays (see input parameter nt)
cr,ci  that contain the vector spherical harmonic coefficients
       of (v,w) as computed by subroutine vhagc.

mdab   the first dimension of the arrays br,bi,cr, and ci as it
       appears in the program that calls vtsgc.

ndab   the second dimension of the arrays br,bi,cr, and ci as it
       appears in the program that calls vtsgc.

wvts   an array which must be initialized by subroutine vtsgci.

lwvts  the dimension of the array wvts as it appears in the
       program that calls vtsgc.

work   a work array that does not have to be saved.

lwork  the dimension of the array work as it appears in the
       program that calls vtsgc.

************************************************************
```

```
      vt,wt  two or three dimensional arrays (see input parameter nt)
             in which the derivative of (v,w) with respect to
             colatitude theta is stored.

             = 1  error in the specification of nlat
             = 2  error in the specification of nlon
             = 3  error in the specification of ityp
             = 4  error in the specification of nt
             = 5  error in the specification of idvw
             = 6  error in the specification of jdvw
             = 7  error in the specification of mdab
             = 8  error in the specification of ndab
             = 9  error in the specification of lwvts
             = 10 error in the specification of lwork
      ****************************************************************
```

vtsgci(...)

```
      ****************************************************************

        subroutine vtsgci initializes the array wvts which can then be
        used repeatedly by subroutine vtsgc until nlat or nlon is change

      ************************************************************

        input parameters

        nlat   the number of gaussian colatitudinal grid points.

        nlon   the number of distinct longitude points.

        lwvts  the dimension of the array wvts as it appears in the
               program that calls vtsgc.

        dwork  a doubleprecision work array that does not have to be sav

        ldwork the dimension of the array dwork as it appears in the
               program that calls vtsgc.

      ************************************************************

        output parameters

        wvts   an array which is initialized for use by subroutine vtsgc

        ierror = 0  no errors
               = 1  error in the specification of nlat
               = 2  error in the specification of nlon
               = 3  error in the specification of lwvts
               = 4  error in the specification of lwork
      ****************************************************************
```

```
     ************************************************************

     given the vector harmonic analysis br,bi,cr, and ci (computed
     by subroutine vhags) of some vector function (v,w), this
     subroutine computes the vector function (vt,wt) which is
     the derivative of (v,w) with respect to colatitude theta. vtsgs
     is similar to vhsgs except the vector harmonics are replaced by
     their derivative with respect to colatitude with the result that
     (vt,wt) is computed instead of (v,w). vt(i,j) is the derivative
     of the colatitudinal component v(i,j) at the gaussian colatitude
     point theta(i) and longitude phi(j) = (j-1)*2*pi/nlon. the
     spectral representation of (vt,wt) is given below at output
     parameters vt,wt.

     ************************************************************

     input parameters

     nlat   the number of gaussian colatitudinal grid points.

     nlon   the number of distinct longitude points.

     ityp   = 0  no symmetries exist about the equator.

     nt     the number of syntheses.

     idvw   the first dimension of the arrays vt,wt as it appears in
            the program that calls vtsgs.

     jdvw   the second dimension of the arrays vt,wt as it appears in
            the program that calls vtsgs. jdvw must be at least nlon.

     br,bi  two or three dimensional arrays (see input parameter nt)
     cr,ci  that contain the vector spherical harmonic coefficients
            of (v,w) as computed by subroutine vhags.

     mdab   the first dimension of the arrays br,bi,cr, and ci as it
            appears in the program that calls vtsgs.

     ndab   the second dimension of the arrays br,bi,cr, and ci as it
            appears in the program that calls vtsgs.

     wvts   an array which must be initialized by subroutine vtsgsi.

     lwvts  the dimension of the array wvts as it appears in the
            program that calls vtsgs.



     work   a work array that does not have to be saved.
```

```
      lwork  the dimension of the array work as it appears in the
             program that calls vtsgs.

      ****************************************************************

      output parameters

      vt,wt  two or three dimensional arrays (see input parameter nt)
             in which the derivative of (v,w) with respect to
             colatitude theta is stored. vt(i,j),wt(i,j) contain the
             derivatives at gaussian colatitude points theta(i).

      ierror = 0  no errors
             = 1  error in the specification of nlat
             = 2  error in the specification of nlon
             = 3  error in the specification of ityp
             = 4  error in the specification of nt
             = 5  error in the specification of idvw
             = 6  error in the specification of jdvw
             = 7  error in the specification of mdab
             = 8  error in the specification of ndab
             = 9  error in the specification of lwvts
             = 10 error in the specification of lwork
      ****************************************************************
```

vtsgsi(...)

```
      ****************************************************************

        subroutine vtsgsi initializes the array wvts which can then be
        used repeatedly by subroutine vtsgs until nlat or nlon is change

      ****************************************************************

        input parameters

        nlat   the number of gaussian colatitudinal grid points.

        nlon   the number of distinct longitude points.

        lwvts  the dimension of the array wvts as it appears in the
               program that calls vtsgs.

        work   a work array that does not have to be saved.

        lwork  the dimension of the array work as it appears in the
               program that calls vtsgs.

        dwork  a doubleprecision work array that does not have to be sav

        ldwork the length of dwork.
```

```
*****************************************************************

  output parameters

  wvts   an array which is initialized for use by subroutine vtsgs

  ierror = 0  no errors
         = 1  error in the specification of nlat
         = 2  error in the specification of nlon
         = 3  error in the specification of lwvts
         = 4  error in the specification of lwork
         = 5  error in the specification of ldwork
*****************************************************************
```

Data

MIRROR = 2
NONE = 0
TRANSPOSE = 1
error = 'spherepack.error'